FDCFIT: A MATLAB Toolbox of Parametric Expressions of the Flow Duration Curve

Jasper A. Vrugt[a,b], and Mojtaba Sadegh

[a]*Department of Civil and Environmental Engineering, University of California Irvine, 4130 Engineering Gateway, Irvine, CA 92697-2175*
[b]*Department of Earth System Science, University of California Irvine, Irvine, CA*

Abstract

The flow duration curve (FDC) is a signature catchment characteristic that depicts graphically the relationship between the exceedance probability of streamflow and its magnitude. This curve is used widely for flood risk analysis, water quality management, and the design of hydroelectric power plants (among others). Several mathematical formulations have been proposed to mimic the FDC. Yet, these functions are often not flexible enough to portray accurately the functional shape of the FDC for a large range of catchments. *Vrugt and Sadegh* (2013) introduced the soil water characteristic (SWC) of van Genuchten (*van Genuchten*, 1980) as new parametric expression of the FDC for diagnostic model evaluation with DREAM$_{(\mathrm{ABC})}$. *Sadegh et al.* (2016) build on the work of *Vrugt and Sadegh* (2013) and compared several models of the SWC against their counterparts published in the literature. These new expressions were shown to fit well the empirical FDCs of the 438 watersheds of the MOPEX data set. Here, we present a MATLAB toolbox, called FDCFIT which contains the fifteen different FDC functions described in *Sadegh et al.* (2016) and returns the values of their coefficients for a given discharge record, along with graphical output of the fit. Two case studies are used to illustrate the main capabilities and functionalities of the FDCFIT toolbox.

Keywords: Flow duration curve, Watershed hydrology, Discharge, Numerical modeling, Diagnostic model evaluation

FDCFIT MANUAL

Email address: jasper@uci.edu (Jasper A. Vrugt)
URL: http://faculty.sites.uci.edu/jasper (Jasper A. Vrugt), http://scholar.google.com/citations?user=zkNXecUAAAAJ&hl=en (Jasper A. Vrugt)

1. Introduction and Scope

The flow duration curve (FDC) is a widely used characteristic signature of a watershed, and is one of the three most commonly used graphical methods in hydrologic studies, along with the mass curve and the hydrograph (*Foster*, 1934). The FDC relates the exceedance probability (frequency) of streamflow to its magnitude, and characterizes both the flow regime and the streamflow variability of a watershed. The FDC is closely related to the "survival" function in statistics (*Vogel and Fennessey*, 1994), and is derived directly from the streamflow cumulative distribution function (CDF). The FDC is frequently used to predict the distribution of streamflow for water resources planning purposes, to simplify analysis of water resources problems, and to communicate watershed behavior to those who lack in-depth hydrologic knowledge. One should be particularly careful however to rely solely on the FDC as main descriptor of catchment behavior (*Vogel and Fennessey*, 1995) as the curve represents the rainfall-runoff as disaggregated in the time domain and hence lacks temporal structure (*Searcy*, 1959; *Vogel and Fennessey*, 1994).

This manual describes a MATLAB toolbox for fitting of the FDC. This toolbox implements the various expressions of *Sadegh et al.* (2016) and returns their optimized coefficients along with graphical output of the quality of the fit. The built-in functions are illustrated using discharge data from two contrasting watersheds in the United States. These example studies are easy to run and adapt and serve as templates for other data sets. The different functions are particularly useful for diagnostic model evaluation as the parameters of each model can be used as summary statistics with the $\text{DREAM}_{(\text{ABC})}$ algorithm (*Vrugt and Sadegh*, 2013; *Sadegh et al.*, 2015, 2016).

The remainder of this manual is organized as follows. Section 2 discusses different parametric expressions of the FDC that are available to the user. This is followed in section 3 with a description of the MATLAB toolbox FDCFIT. In this section we are especially concerned with the input and output arguments of FDCFIT and the various utilities and options available to the user. Section 4 discusses two case studies which illustrate how to use the toolbox. The penultimate section of this paper (section 5) highlights recent research efforts aimed at further improving the fitting of flow duration curves with specific emphasis on the mathematical description of their spatial variability using the scaling framework. Finally, section 6 concludes this manual with a summary of the main findings.

Note, other toolboxes developed by the first author of this manual include the DREAM algorithm for Bayesian inference (*Vrugt*, 2016) , the AMALGAM method for multiple objective optimization (*Vrugt*, 2015a), and MODELAVG for model averaging (*Vrugt*, 2015c).

2. Inference of the Flow Duration Curve

The flow duration curve (FDC) is a signature catchment characteristic that depicts graphically the relationship between the exceedance probability of streamflow and its magnitude. Figure 1 shows the empirical (observed) FDCs of eight watersheds of the MOPEX data set. Note that the streamflow values on the y-axis are normalized so that different watersheds are more easily compared.

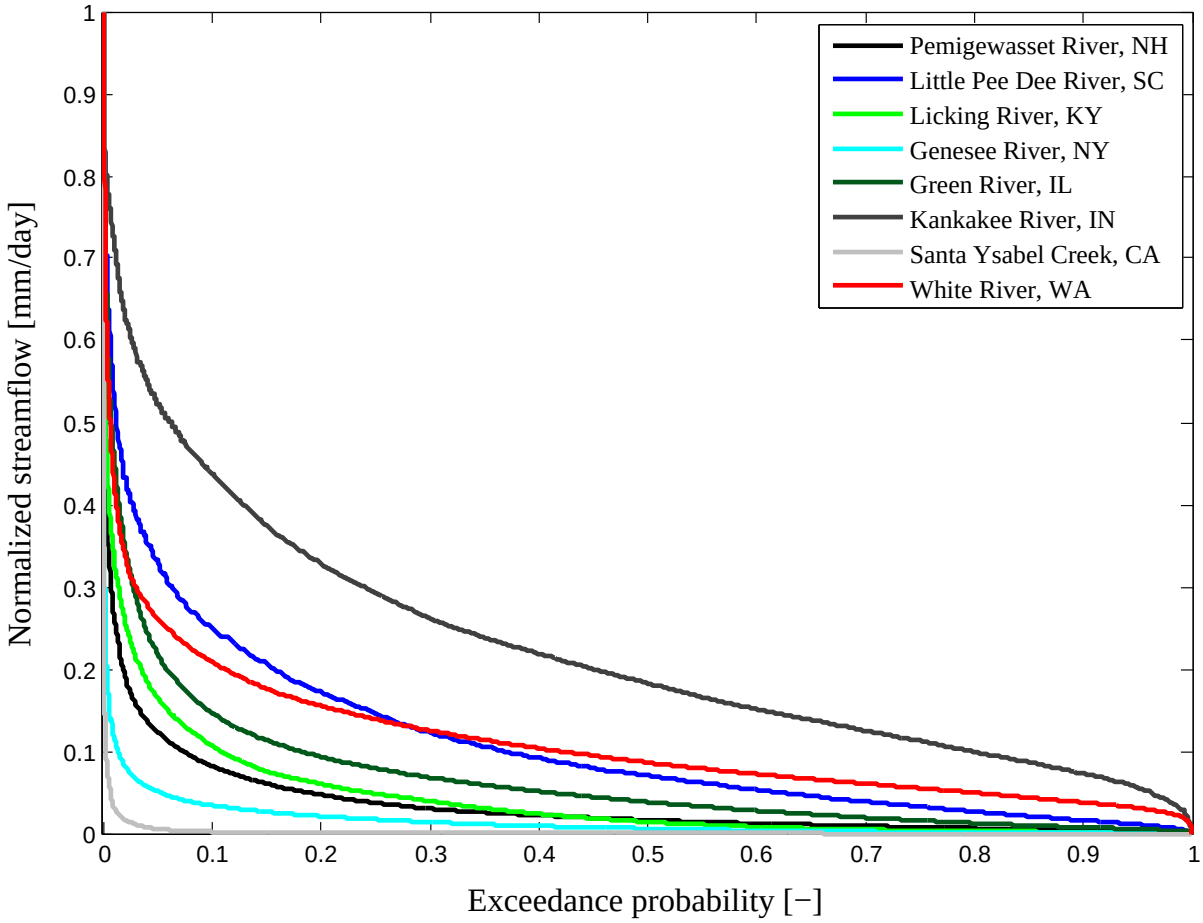


Figure 1: Flow duration curves of eight watersheds of the MOPEX data set. The watersheds exhibit quite contrasting hydrologic behavior - some retain water much better than others and hence the discharge response to rainfall is more delayed. The streamflow values are normalized so that the FDCs of the different watersheds are easily compared.

The plotted FDCs differ quite substantially from each other - a reflection of differences among the watersheds in their transformation of rainfall into runoff emanating from the catchment outlet. Ideally, we would have available a single parametric expression that can fit very closely the empirical FDCs of each of these watersheds.

2.1. The exceedance probability

In this section we briefly describe how one can compute exceedance probabilities of a data record of a certain entity. Different methods have been proposed in the statistical literature to do so, yet they provide similar results if the data record is sufficient long.

If we denote with $\hat{\mathbf{Y}} = \{\hat{y}_1, \ldots, \hat{y}_n\}$ the observed record of n discharge values, then we can calculate the exceedance probability, $\hat{\mathbf{E}} = \{\hat{e}_1, \ldots, \hat{e}_n\}$, of each value of $\hat{\mathbf{Y}}$ as follows. We first sort the n-vector of $\hat{\mathbf{Y}}$ in descending order and store vector of discharge values in the vector $\hat{\mathbf{Y}}_{rms}$. Each element of this sorted vector is then assigned a rank, $\mathbf{R} = \{1, \ldots, n\}$ starting with one for the largest discharge value (first element of

$\hat{\mathbf{Y}}_{rms}$). We can now calculate the exceedance probability, e_i of each discharge observation, $i = \{1, \ldots, n\}$ of $\hat{\mathbf{Y}}_{rms}$ using the Weibull plotting position

$$\hat{e}_i = \frac{1}{n}(r_i - \frac{1}{2}), \tag{1}$$

where i denotes the element number of $\hat{\mathbf{Y}}_{\mathrm{s}}$ (equivalent to rank), and $\hat{\mathbf{E}} = \{\hat{e}_1, \ldots, \hat{e}_n\}$ signifies the n-vector of corresponding exceedance probabilities. Other formulations of Equation (1) have been proposed in the statistical literature but provide very similar estimates of the exceedance probabilities for large data records, say $n > 100$.

2.2. Semi-arid watersheds

The FDC is an important signature of the catchment response to rainfall and is relatively easy to construct from the observed discharge record. It only requires a function to sort the discharge data record. One issue, however deserves special attention, and that is the presence of (near)-zero flows. This is common for (dry-bed) ephemeral or intermittent streams in semi-arid watersheds that alternate between short flash-flood events characterized by pronounced runoff dynamics and rapidly rising hydrographs, and long periods of (nearly) zero flows. With treatment of zero flow days the FDC would consist of two different parts, a regular "S"-shaped section between values of the exceedance probability, $\hat{e}_i \in [0, 1 - \hat{p}_0]$ and $\hat{y} > 0$, and a horizontal portion for $\hat{e}_i \in [1 - \hat{p}_0, 1]$ and $\hat{y} = 0$, where $\hat{p}_0$ denotes the probability of zero flows. The value of $\hat{p}_0$ is easily computed from the observed record of discharge values

$$\hat{p}_0 = \frac{n_0}{n} \tag{2}$$

where n_0 signifies the number of zero flow observations. The value of $\hat{p}_0$ is not as much determined by watershed properties (geology, soils, slope, etc.) but rather by climatic conditions (precipitation). We therefore assume $\hat{p}_0$ to be known for each watershed and use the following variable instead

$$\hat{e}_i^* = \frac{\hat{e}_i}{1 - \hat{p}_0}, \tag{3}$$

in each parametric expression of the FDC. This transformed variable scales linearly the exceedance probabilities of the non-zero flows to the interval $[0, 1]$. The exceedance probabilities of the original discharge record are then easily derived by inverting Equation (3). Note, the majority of the MOPEX data records ($> 82\%$) has strictly positive flows, and thus for those watersheds $\hat{p}_0 = 0$ and consequently $\hat{e}_i^* = \hat{e}_i$. We now use the wording "empirical FDC" to denote the n measured $\{\hat{e}_i, \hat{y}_i\}$ pairs of the watershed.

2.3. Probabilistic and mathematical functions

We now review a suite of different functions commonly used in the hydrologic literature to mimic the empirical FDC. These functions can be grouped in two main classes. The first class builds on commonly used probability distributions, and are also referred to in the literature as probabilistic models. The second class of models uses standard parametric expressions to capture the "S"-shaped curve of the FDC. Unlike the first class of models which uses the mathematical equation of the (inverse) CDF, this second group of nonprobabilistic models involves a higher-degree of trial-and-error in their development.

2.3.1. Probabilistic models

This class of models is used widely by researchers and practitioners in large part due to their relative parametric simplicity, flexibility, solid statistical underpinning and relative easy of derivation from the cumulative distribution function (CDF). Indeed, the probability of exceedence, e, is easily computed from the cumulative distribution function as follows

$$e = 1 - p(y \leq Y) = 1 - \mathrm{CDF}(y), \tag{4}$$

so that the highest flows (CDF close to unity) are associated with an exceedance probability of zero. Thus, any statistical distribution with a closed-form mathematical expression for its CDF is a viable candidate for the FDC. In the remainder of this section, we conveniently refer to the transformation of Equation (4) as the pseudo-CDF or pseudo-inverse CDF.

The first of these models involves the Gumbel distribution (*Booker and Snelder*, 2012). The skew of the discharge data, so clearly visible in a frequency distribution of the streamflows, prompts the use of this model. The pseudo-inverse CDF of the Gumbel distribution is given by

$$y_i = a_{\mathrm{G}} - b_{\mathrm{G}} \log\left[-\log\left(1 - e_i^*\right)\right], \tag{5}$$

where e_i^* denotes the scaled exceedance probability, and a_{G} (mm/day) and b_{G} (mm/day) are coefficients that define the location and scale of the Gumbel distribution. Equation (5) is a special case of the generalized extreme value (GEV) distribution. The pseudo-inverse CDF of this distribution is yet another parametric formulation for the empirical FDC

$$y_i = a_{\mathrm{GEV}} + \frac{b_{\mathrm{GEV}}}{c_{\mathrm{GEV}}}\left[\left(-\log(1 - e_i^*)\right)^{-c_{\mathrm{GEV}}} - 1\right], \tag{6}$$

where a_{GEV} (mm/day), b_{GEV} (mm/day), and c_{GEV} (-) denote the location, scale, and shape parameters of the GEV distribution, respectively. The shape parameter, c_{GEV}, controls the skewness of the distribution and enables the fitting of tailed streamflow distributions. The GEV distribution is widely used in the field of hydrology to model floods and drought (extremes) (*Katz et al.*, 2002), as well as the FDC (*Booker and Snelder*, 2012). If $c_{\mathrm{GEV}} = 0$ then the GEV distribution simplifies to the Gumbel distribution.

The lognormal distribution is another statistical distribution that is used widely for modeling of the FDC [*Fennessey and Vogel* (1990), among others]. This model can produce the skew necessary to describe the non-symmetrical shape of the frequency distribution of streamflows. The pseudo-inverse CDF of the lognormal distribution is given by

$$y_i = \exp\left[a_{\mathrm{LN}} - \sqrt{2}\, b_{\mathrm{LN}}\, \mathrm{erfc}^{-1}\left(2(1 - e_i^*)\right)\right], \tag{7}$$

where $\mathrm{erfc}^{-1}(x)$ returns the value of the inverse complementary error function evaluated at x, and a_{LN} (mm/day) and b_{LN} (mm/day) are location and scale coefficients of the lognormal distribution, respectively. Several others have shown that a third parameter can improve signifincantly the fit of this model to the empirical FDC [*Longobardi and Villani* (2013), among others]. This results in the following expression for

the pseudo-inverse CDF of the 3-parameter lognormal distribution

$$y_i = c_{\mathrm{LN}} + \exp\left[a_{\mathrm{LN}} - \sqrt{2}\, b_{\mathrm{LN}}\, \mathrm{erfc}^{-1}\left(2(1-e_i^*)\right)\right], \tag{8}$$

where c_{LN} (mm/day) signifies the additional coefficient. If $c_{\mathrm{LN}} = 0$ this function simplifies to the 2-parameter formulation of the lognormal distribution.

We are now left with two remaining statistical distributions whose CDF can be used to describe the empirical FDC. This involves the logistic and the generalized Pareto distribution. These distributions have not enjoyed widespread application and use in the hydrologic literature, but we consider their application herein to complete the set of probabilistic models. The pseudo-inverse CDF of the logistic distribution (LG) is given by

$$y_i = a_{\mathrm{LG}} - b_{\mathrm{LG}} \log\left(\frac{1}{1-e_i^*} - 1\right), \tag{9}$$

where a_{LG} (mm/day) and b_{LG} (mm/day) are coefficients of the distribution. The pseuo-inverse CDF of the generalized Pareto (GP) distribution is defined as follows

$$y_i = a_{\mathrm{GP}} + \frac{b_{\mathrm{GP}}}{c_{\mathrm{GP}}}\left[(e_i^*)^{-c_{\mathrm{GP}}} - 1\right], \tag{10}$$

and the fitting coefficients a_{GP} (mm/day), b_{GP} (mm/day), and c_{GP} (-) need to be derived by fitting against the empirical FDC. This concludes the description of the probabilistic functions (models) of the FDC. In a later section of this paper we will detail how the coefficients in each of these functions are derived. The next section continues our review with several nonprobabilistic models of the FDC. These models do not enjoy a rigorous statistical footing and are typically derived by trial-and-error.

2.3.2. Nonprobabilistic models

The second class of FDC models used in the literature builds on rather standard parametric expressions. These models have in common with their probabilistic counterparts the use of fitting coefficients. These coefficients are subject to inference using some nonlinear minimization method and data of the empirical FDC. The first two nonprobabilistic models involve the logarithmic (LOG) and power (PW) function

$$y_i = b_{\mathrm{LOG}} + a_{\mathrm{LOG}} \log(e_i^*), \tag{11}$$

and

$$y_i = b_{\mathrm{PW}} (e_i^*)^{-a_{\mathrm{PW}}}, \tag{12}$$

where a_{LOG} (mm/day), b_{LOG} (mm/day), a_{PW} (-), and b_{PW} (mm/day) are fitting coefficients. A 2-parameter exponential function was suggested by *Quimpo et al.* (1983)

$$y_i = a_{\mathrm{Q}}\, \exp(-b_{\mathrm{Q}} e_i^*), \tag{13}$$

with coefficients a_{Q} (mm/day) and b_{Q} (-).

Franchini and Suppo (1996) proposed a 3-parameter expression of the FDC defined as

$$y_i = b_{\mathrm{FS}} + a_{\mathrm{FS}}(1 - e_i^*)^{c_{\mathrm{FS}}}, \tag{14}$$

in which a_{FS} (mm/day), b_{FS} (mm/day) and c_{FS} (-) denote the fitting coefficients. This parametric function was originally proposed to describe only the low flow part of the FDC, but has been applied to the entire FDC as well (*Sauquet and Catalogne*, 2011).

More recently, *Viola et al.* (2011) proposed a simple 2-parameter function which is given by

$$y_i = \begin{cases} a_{\mathrm{V}} \left(\frac{1}{e_i^*} - 1\right)^{b_{\mathrm{V}}} & \text{if } e_i < 1 - e_0 \\ \\ 0 & \text{if } e_i \geq 1 - e_0 \end{cases}, \tag{15}$$

and a_{V} (mm/day), and b_{V} (-) are the fitting coefficients. This concludes our description of the nonprobabilistic FDC models.

Practical experience suggests that the probabilistic and nonprobabilistic models discussed thus far are not always flexible enough to fit closely the FDC for a large range of watersheds with contrasting hydrologic behaviors. *Vrugt and Sadegh* (2013) therefore introduced a new class of parametric expressions for the FDC which mimic closely the FDC of the MOPEX data set and build on commonly used models of the soil water retention function. *Sadegh et al.* (2016) provides an in-depth analysis of these new formulations and compared their performance against existing models used in the literature. The next section describes these new models.

2.4. Proposed parametric formulations

The functional shape of the FDC has many elements in common with that of the soil water characteristic (SWC). This is graphically illustrated in Figure 2 which plots the water retention function of five different soils presented in *van Genuchten* (1980). These curves depict the relationship between the volumetric moisture content, θ (x-axis) and the corresponding pressure head, h (y-axis) of a soil and are derived by fitting the following equation

$$\theta = \theta_r + (\theta_s - \theta_r)\left[1 + (\alpha|h|)^n\right]^{-m}, \tag{16}$$

to experimental (θ, h) data collected in the laboratory. This equation is also known as the van Genuchten (VG) model and contains five different parameters, where θ_s (cm^3/cm^3) and θ_r (cm^3/cm^3) denote the saturated and residual moisture content, respectively, and α (1/cm), n (-) and m (-) are fitting coefficients that determine the air-entry value and slope of the SWC. In most studies, the value of m is set conveniently to $1-1/n$ which not only reduces the number of parameters to four, but also provides a closed-form expression for the unsaturated soil hydraulic conductivity function (*van Genuchten*, 1980).

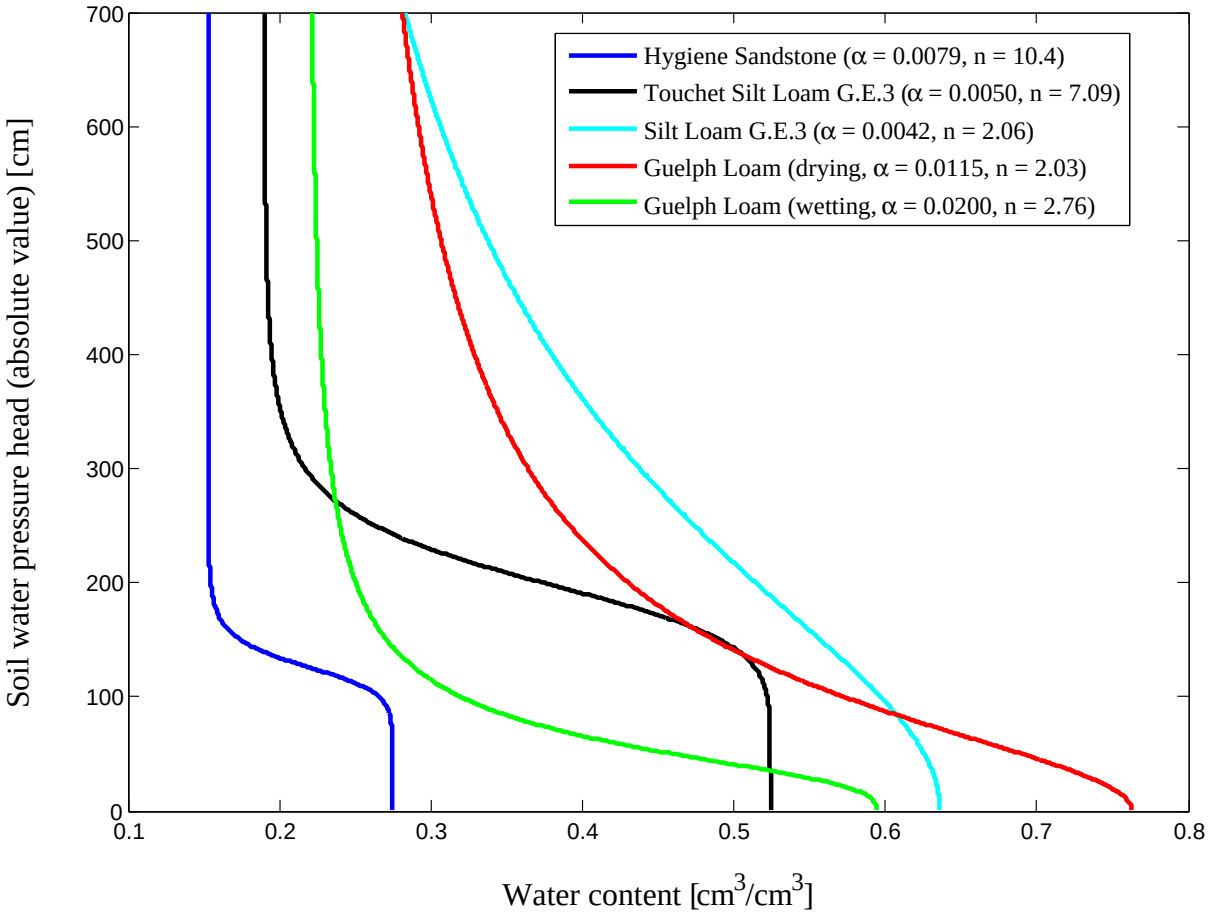


Figure 2: Water retention functions of five different soil types (derived from *van Genuchten* (1980)). This curve depicts the relationship between the water content, θ (cm^3/cm^3), and the soil water potential, h (cm). This curve is characteristic for different types of soil, and is also called the soil moisture characteristic.

The shape of the WRFs plotted in Figure 2 show great similarity with the FDCs displayed previously in Figure 1. This suggests that Equation (16) might be a good parametric expression to describe the FDC and thus relationship between the exceedance probability of streamflow (x-axis) and its magnitude (y-axis). To make sure that the exceedance probability is bounded exactly between 0 and 1, we set $\theta_r = 0$ and $\theta_s = 1$, respectively. This leads to the following 3-parameter VG formulation of the FDC proposed by *Vrugt and Sadegh* (2013)

$$e_i^* = \left[1 + (a_{\mathrm{VG}} y_i)^{b_{\mathrm{VG}}}\right]^{-c_{\mathrm{VG}}}, \tag{17}$$

with coefficients a_{VG} (day/mm), b_{VG} (-), and c_{VG} (-). This VG-formulation relates the streamflow (input variable) to its exceedance probability (output variable), and has to be inverted to be consistent with the other formulations used herein. This gives

$$y_i = \frac{1}{a_{\mathrm{VG}}} \left((e_i^*)^{(-1/c_{\mathrm{VG}})} - 1 \right)^{(1/b_{\mathrm{VG}})}. \tag{18}$$

The VG-formulation presented in Equation (18) can be simplified to a 2-parameter function if we assume that $c_{\mathrm{VG}} = 1 - 1/b_{\mathrm{VG}}$. We will consider both VG formulations of the FDC.

The VG model is used widely in porous flow simulators to describe numerically variable unsaturated water flow. Yet, many other hydraulic models have been proposed in the vadose zone literature to characterize the retention and unsaturated soil hydraulic conductivity functions. We consider here the lognormal WRF of *Kosugi* (1994, 1996)

$$e_i^* = \begin{cases} \frac{1}{2}\,\mathrm{erfc}\left[\frac{1}{\sqrt{2}b_{\mathrm{K}}}\log\left(\frac{y_i - c_{\mathrm{K}}}{a_{\mathrm{K}} - c_{\mathrm{K}}}\right)\right] & \text{if } y_i > c_{\mathrm{K}} \\ 1 & \text{if } y_i \leq c_{\mathrm{K}} \end{cases}, \tag{19}$$

where a_{K} (mm/day), b_{K} (-) and c_{K} (mm/day) are the coefficients of the Kosugi model that need to be determined by calibration against the empirical FDC of each watershed. We now need to invert Equation (19) to get as output the exceedance probability for a given value of the streamflow

$$y_i = c_{\mathrm{K}} + (a_{\mathrm{K}} - c_{\mathrm{K}})\,\exp\left(\sqrt{2}\,b_{\,\mathrm{K}}\,\mathrm{erfc}^{-1}(2g_i^*)\right). \tag{20}$$

We can simplify this 3-parameter formulation by setting $c_{\mathrm{K}} = 0$. This 2-parameter formulation of Kosugi is then, after a log-transformation and some rearrangement, equivalent mathematically to the 2-parameter lognormal distribution. The 3-parameter Kosugi model differs however from its counterpart of the lognormal distribution.

The main advantage of the WRF of Kosugi is that its parameters can be related directly to the pore size distribution and hence exhibit a much better physical underpinning than their counterparts of the VG model. This might increase the chances of a successful regionalization.

A summary of the different FDC models appears in Table 1. These models are part of the FDCFIT toolbox that will be discussed in the next section.

Table 1: Summary of the probabilistic, nonprobabilistic, and proposed WRF-based FDC models. All these functions return the streamflow value, y_i, for a given value of the exceedance probability, e_i^*. The variables a, b (models [1] - [10]), and a, b, and c (models [11] - [15]) are fitting coefficients whose values are derived from the empirical FDC.

Model name	No.	Mathematical expression, $y = \mathcal{F}(e)$
Lognormal	[1]	$y_i = \exp\left[a_{\mathrm{LN}} - \sqrt{2}\, b_{\mathrm{LN}}\, \mathrm{erfc}^{-1}\left(2(1-e_i^*)\right)\right]$
Gumbel	[2]	$y_i = a_{\mathrm{G}} - b_{\mathrm{G}}\, \log\left[-\log\left(1-e_i^*\right)\right]$
Logistic	[3]	$y_i = a_{\mathrm{LG}} - b_{\mathrm{LG}}\, \log\left(\frac{1}{1-e_i^*} - 1\right)$
Logarithmic	[4]	$y_i = b_{\mathrm{LOG}} + a_{\mathrm{LOG}}\, \log(e_i^*)$
Power	[5]	$y_i = b_{\mathrm{PW}}(e_i^*)^{-a_{\mathrm{PW}}}$
Quimpo	[6]	$y_t = a_{\mathrm{Q}}\, \exp\left(-b_{\mathrm{Q}} e_i^*\right)$
Viola	[7]	$y_i = a_{\mathrm{V}} \left(\frac{1}{e_i^*} - 1\right)^{b_{\mathrm{V}}}$
van Genuchten	[8]	$y_i = \frac{1}{a_{\mathrm{VG}}} \left[(e_i^*)^{1/(1-1/b_{\mathrm{VG}})} - 1\right]^{\frac{1}{b_{\mathrm{VG}}}}$
Kosugi	[9]	$y_i = a_{\mathrm{K}}\, \exp\left(\sqrt{2}\, b_{\mathrm{K}}\, \mathrm{erfc}^{-1}(2e_i^*)\right)$
Lognormal	[10]	$y_i = c_{\mathrm{LN}} + \exp\left[a_{\mathrm{LN}} - \sqrt{2}\, b_{\mathrm{LN}}\, \mathrm{erfc}^{-1}\left(2(1-e_i^*)\right)\right]$
Generalized Pareto	[11]	$y_i = a_{\mathrm{GP}} + \frac{b_{\mathrm{GP}}}{c_{\mathrm{GP}}} \left[(e_i^*)^{-c_{\mathrm{GP}}} - 1\right]$
Extreme Value	[12]	$y_i = a_{\mathrm{GEV}} + \frac{b_{\mathrm{GEV}}}{c_{\mathrm{GEV}}} \left\{\left[-\log(1-e_i^*)\right]^{-c_{\mathrm{GEV}}} - 1\right\}$
Franchini and Suppo	[13]	$y_t = b_{\mathrm{FS}} + a_{\mathrm{FS}}(1-e_i^*)^{c_{\mathrm{FS}}}$
van Genuchten	[14]	$y_i = \frac{1}{a_{\mathrm{VG}}} \left[(e_i^*)^{-1/c_{\mathrm{VG}}} - 1\right]^{(1/b_{\mathrm{VG}})}$
Kosugi	[15]	$y_i = c_{\mathrm{K}} + (a_{\mathrm{K}} - c_{\mathrm{K}})\, \exp\left[\sqrt{2}\, b_{\mathrm{K}}\, \mathrm{erfc}^{-1}\left(2e_i^*\right)\right]$

2.5. Change of dependent/independent variables

The functions of the FDC presented in the previous section return the streamflow value (dependent variable) for a given exceedance probability (independent variable), or $y = \mathcal{F}(e)$. For practical considerations, however it might be useful to have available a direct expression for the exceedance probability instead, and thus $e = \mathcal{F}^{-1}(y)$. This inverse formulation returns the exceedance probability (dependent variable) for a given streamfow value (independent variable) and is particularly useful if one wants to compute (among others) the relative amount of time that the streamflow is likely to exceed a certain target. If this flow value constitutes the maximum capacity of the channel, then the probability of flooding can be assessed. Indeed, this inverse formulation is of great value to decision makers concerned with the design and engineering of dams and other flood protection structures. For example, a structure can be designed to perform well within some range of flows, such as flows that occur between 20 and 80% of the time (or some other selected interval).

The inverse formulation, $e = \mathcal{F}^{-1}(y)$ is rather straightforward to derive for each of the FDC models listed in the previous section. For example, the inverse of the Gumbel distribution in Equation (5) can be derived in the following few steps. We first note that $e_i^* = \mathcal{F}^{-1}(y_i)$ and replace the exceedance probability, e_i^* in Equation (5) with $\mathcal{F}^{-1}(y_i)$. This gives

$$y_i = a_{\mathrm{G}} - b_{\mathrm{G}} \log\left[-\log\left(1 - \mathcal{F}^{-1}(y_i)\right)\right]. \tag{21}$$

We now isolate $\mathcal{F}^{-1}(y_i)$

$$\log\left[-\log\left(1 - \mathcal{F}^{-1}(y_i)\right)\right] = \frac{a_{\mathrm{G}} - y_i}{b_{\mathrm{G}}}, \tag{22}$$

and get rid of the first $\log(\cdot)$ operator at the left hand side using the exponential function

$$-\log(1-\mathcal{F}^{-1}(y_i)) = \exp\left(\frac{a_{\mathrm{G}}-y_i}{b_{\mathrm{G}}}\right). \tag{23}$$

If we repeat the same operation, and rearrange the final equation we derive

$$e_i^* = \mathcal{F}^{-1}(y_i) = 1-\exp\left[-\exp\left(\frac{a_{\mathrm{G}}-y_i}{b_{\mathrm{G}}}\right)\right]. \tag{24}$$

These steps can be repeated for each of the parametric expressions of the FDC in the previous section 2.3.

Table 2: Inverse formulation, $\mathcal{F}^{-1}(y)$ of each FDC model of Table 1. These functions return the exceedance probability, e_i^*, for a given value of the streamflow, y_i. The variables a, b (models [1] - [10]), and a, b, and c (models [11] - [15]) are fitting coefficients whose values are derived from the empirical FDC.

Model name	No.	Mathematical expression, $e = \mathcal{F}^{-1}(y)$
Lognormal	[1]	$e_i^* = 1-\frac{1}{2}\mathrm{erfc}\left(\frac{a_{\mathrm{LN}}-\log(y_i)}{\sqrt{2}\,b_{\mathrm{LN}}}\right)$
Gumbel	[2]	$e_i^* = 1-\exp\left[-\exp\left(\frac{a_{\mathrm{G}}-y_i}{b_{\mathrm{G}}}\right)\right]$
Logistic	[3]	$e_i^* = 1-\left[1+\exp\left(\frac{a_{\mathrm{LG}}-y_i}{b_{\mathrm{LG}}}\right)\right]^{-1}$
Quimpo	[4]	$e_i^* = -\frac{1}{b_{\mathrm{Q}}}\log\left(\frac{y_i}{a_{\mathrm{Q}}}\right)$
Viola	[5]	$e_i^* = \left[\left(\frac{y_i}{a_{\mathrm{V}}}\right)^{(1/b_{\mathrm{V}})}+1\right]^{-1}$
Logarithmic	[6]	$e_i^* = \exp\left(\frac{y_i-b_{\mathrm{LOG}}}{a_{\mathrm{LOG}}}\right)$
Power	[7]	$e_i^* = \left(\frac{y_i}{b_{\mathrm{PW}}}\right)^{(-1/a_{\mathrm{PW}})}$
van Genuchten	[8]	$e_i^* = \left[1+(a_{\mathrm{VG}}y_i)^{b_{\mathrm{VG}}}\right]^{(1/b_{\mathrm{VG}}-1)}$
Kosugi	[9]	$e_i^* = \frac{1}{2}\,\mathrm{erfc}\left[\frac{1}{\sqrt{2}\,b_{\mathrm{K}}}\log\left(\frac{y_i}{a_{\mathrm{K}}}\right)\right]$
Lognormal	[10]	$e_i^* = \begin{cases} 1-\frac{1}{2}\,\mathrm{erfc}\left[\frac{a_{\mathrm{LN}}-\log(y_i-c_{\mathrm{LN}})}{\sqrt{2}\,b_{\mathrm{LN}}}\right] & \text{if} \quad y_i > c_{\mathrm{LN}} \\ 1 & \text{if} \quad y_i \le c_{\mathrm{LN}} \end{cases}$
Generalized Pareto	[11]	$e_i^* = 1-\begin{cases} \left[1+\frac{c_{\mathrm{GP}}(y_i-a_{\mathrm{GP}})}{b_{\mathrm{GP}}}\right]^{(-1/c_{\mathrm{GP}})} & \text{if} \quad c_{\mathrm{GP}} \neq 0 \\ \exp\left(\frac{a_{\mathrm{GP}}-y_i}{b_{\mathrm{GP}}}\right) & \text{if} \quad c_{\mathrm{GP}} = 0 \end{cases}$
Extreme Value	[12]	$e_i^* = \begin{cases} 1-\exp\left\{-\left[1+c_{\mathrm{GEV}}\left(\frac{y_i-a_{\mathrm{GEV}}}{b_{\mathrm{GEV}}}\right)\right]^{(-1/c_{\mathrm{GEV}})}\right\} & \text{if} \quad c_{\mathrm{GEV}} \neq 0 \\ 1-\exp(-\exp(\left(\frac{a_{\mathrm{GP}}-y_i}{b_{\mathrm{GP}}}\right) & \text{if} \quad c_{\mathrm{GEV}} = 0 \end{cases}$
Franchini and Suppo	[13]	$e_i^* = \left[1-\left(\frac{y_i-b_{\mathrm{FS}}}{a_{\mathrm{FS}}}\right)^{(1/c_{\mathrm{FS}})}\right]$
van Genuchten	[14]	$e_i^* = \left[1+(a_{\mathrm{VG}}y_i)^{b_{\mathrm{VG}}}\right]^{-c_{\mathrm{VG}}}$
Kosugi	[15]	$e_i^* = \begin{cases} \frac{1}{2}\,\mathrm{erfc}\left[\frac{1}{\sqrt{2}\,b_{\mathrm{K}}}\log\left(\frac{y_i-c_{\mathrm{K}}}{a_{\mathrm{K}}-c_{\mathrm{K}}}\right)\right] & \text{if} \quad y_i > c_{\mathrm{K}} \\ 1 & \text{if} \quad y_i \le c_{\mathrm{K}} \end{cases}$

Table 2 lists the inverse formulations, $e = \mathcal{F}^{-1}(y)$ of each of the models listed in Table 1. These models compute the exceedance probability (dependent variables) as function of the streamflow (independent variable).

2.6. Parameter estimation of FDC models

We have developed a MATLAB toolbox called "FDCFIT" which implements the forward and inverse formulation of each of the FDC parametric expressions listed in Table 1 and automatically determines the values of their coefficients from the empirical FDC. Graphical output is provided as well. A sum of squared error (SSE) objective function is used to quantify the distance between the simulated and observed FDC

$$\begin{aligned} \mathrm{SSE}(\mathbf{x}|\hat{\mathbf{E}}) &= \sum_{t=1}^{n} \left(\hat{y}_t - \mathcal{F}(\mathbf{x}|\hat{e}_t)\right)^2 \qquad \text{in } \mathbf{Y}\text{-space} \\ \mathrm{SSE}(\mathbf{x}|\hat{\mathbf{Y}}) &= \sum_{t=1}^{n} \left(\hat{e}_t - \mathcal{F}^{-1}(\mathbf{x}|\hat{y}_t)\right)^2 \quad \text{in } \mathbf{E}\text{-space} \end{aligned} \tag{25}$$

where $\mathbf{x} = \{a., b.\}$ or $\{a., b., c.\}$ signifies the m-vector of fitting coefficients, $\hat{\mathbf{E}} = \{\hat{e}_1, \ldots, \hat{e}_n\}$ and $\mathcal{F}(\cdot)$ and $\mathcal{F}^{-1}(\cdot)$ signify the formulations of the FDC model in the streamflow (Table 1) and exceedance probability (Table 2) space.

These two calibration cases are necessary to fit the forward and inverse formulation of each model to the empirical FDC. Experience suggests that calibration in the $\mathbf{Y}$-space (forward formulation) emphasizes fitting of the peak flows at low exceedance probabilities (largest residuals are found at highest flows), whereas calibration in the $\mathbf{E}$-space (inverse formulation) places equal importance on each observation of the exceedance probability, and therefore should lead to a calibrated model that mimics nicely the entire FDC. Note that a ℓ_1-type objective function would increase the sensitivity of the forward formulations, $y = \mathcal{F}(e)$ to the lower flows

$$\ell_1 = \sum_{t=1}^{n} \left\{\mathrm{abs}\big(\hat{y}_t - \mathcal{F}(\mathbf{x}|\hat{e}_t)\big)\right\} \qquad \text{in } \mathbf{Y}\text{-space.} \tag{26}$$

as this metric no longer squares the distance (residuals) between the observed and predicted streamflow values but rather simply takes their sum. This should promote the fitting of the middle and right tail of the FDC.

Unfortunately, no analytic solution exists for the minimization of the SSE metric in Equation (25). This necessitates the use of iterative solution methods that minimize the SSE in a series of steps. The use of such methods would seem rather trivial as the FDC models of Table (1) only contain two or three parameters. Practical experience suggest however that it is not particularly easy to estimate the model parameters from an empirical FDC. Indeed, in *Sadegh et al.* (2016) we have witnessed for quite a few watersheds and models that the SSE response surface can have large flat areas with a global optimum that is perfectly hidden in a very small pocket of attraction close to the (lower) bound of one of the parameters. This makes it very difficult for optimization methods to locate rapidly the global minimum of the parameter space. Indeed, a comparison of a suite of different optimization algorithms for the 438 basins of the MOPEX data set has shown that all these methods suffer from premature convergence for at least a few FDCs. This makes FDC fitting difficult as we cannot always trust the results of a certain optimization algorithm.

The FDCFIT toolbox implements four different optimization algorithms to calibrate the parameters of each FDC model against data. This includes the multistart Levenberg-Marquardt (*Marquardt*, 1963) and Nelder-Mead, or down-hill Simplex, method (*Nelder and Mead*, 1965), Differential Evolution (*Storn and Price*, 1997) and the Covariance Matrix Adaptation Evolutionary Strategy, CMA (*Hansen and Ostermeier*,

2001). The first two of these four approaches are classified as local optimization methods and improve iteratively the parameter values starting from a single, randomly chosen, point on the response surface. Multiple different (independent) trials are performed to enhance convergence to the global minimum. The last two methods constitutes global optimization algorithms and evolve multiple different solutions (parameter vectors) simultaneously using evolutionary principles of survival of the fittest. Whichever method is used, one should be careful not to place too much confidence in the estimated parameters. We would therefore advise to compare the results of the different optimization algorithms.

3. FDCFIT toolbox in MATLAB

The basic code of FDCFIT was written in 2014 and some changes have been made recently to better support the needs of users. You can download the FDCFIT toolbox from my website at the following link `http://faculty.sites.uci.edu/FDCFIT`. Appendix A explains how to setup the FDCFIT toolbox in MATLAB

The FDCFIT code can be executed from the MATLAB prompt by the following command

```
[x,RMSE] = FDCFIT(FDCPar,E,Y,optim,options)
```
(27)

where `FDCPar` (structure array), `E` ($n \times 1$ vector), `Y` ($n \times 1$ vector), `optim` (structure array) and `options` (structure array) are input arguments defined by the user, and `x` (vector) and `RMSE` (scalar) are output variables computed by FDCFIT and returned to the user. To minimize the number of input and output arguments in the FDCFIT function call and related primary and secondary functions called by this program, we use MATLAB structure arrays and group related variables in one main element using data containers called fields, more of which later. The fourth and fifth input argument of FDCFIT, the structures `optim` and `options` are optional. Default values will be assumed for their fields if both arguments are not defined in the call to FDCFIT. We will now discuss the content and usage of each input argument.

The functions that are used in the MATLAB package of FDCFIT are briefly summarized in Appendix A. In the subsequent sections we will discuss the MATLAB implementation of FDCFIT. This, along with a few prototype case studies presented herein and template examples listed in RUNFDCFIT should help users apply the toolbox to their own data set and models (for diagnostic model evaluation).

3.1. Input argument 1: `FDCPar`

The structure `FDCPar` is used to communicate to FDCFIT the FDC model formulation and the optimization method that is to be used to calibrate its parameters. Table 3 lists the three different fields of `FDCPar`, their possible entries, variable type, and default settings (values).

Table 3: Content of the first input argument `FDCPar` of the main program FDCFIT of the MATLAB toolbox. Each row signifies a different field of `FDCPar` and summarizes name, content, options, and variable type.

Field `FDCPar`	Description	Options	Type
model	model used	$[1, 2, \ldots, 15]$	integer
formulation	formulation	'Y' / 'E'	string
method	optimization algorithm	'LM'/'SP'/'DE'/'CMA'	string

The fields model and formulation of structure FDCPar determine together which mathematical expression should be used to mimic the empirical FDC. The field model contains an integer (1-15) which corresponds directly to the values listed in Tables 1 and 2 for each of the different FDC models. For instance, if model = 12, then the user has selected as FDC model the GEV distribution. This field does not yet specify however which of the two formulations of this model is used - the forward (Table 1) or the inverse expression (Table 2).

The field formulation contains a string which defines with formulation of model should be used. This can either be the forward expression, $y = \mathcal{F}(e)$ (see Table 1) or the inverse formulation, $e = \mathcal{F}^{-1}(y)$ listed in Table 2. This choice determines the calibration results as calibration in the $\mathbf{Y}$-space emphasizes fitting of the peak flows at low exceedance probabilities, and calibration in the $\mathbf{E}$-space places equal weight on all flow levels. Note that we can use the ℓ_1-type objective function of Equation (26) to reduce sensitivity to the large flows when fitting in the $\mathbf{Y}$-space. The current version of the toolbox does not implement this alternative fitting criterion, yet it is easy to implement as it requires only a simple modification of one line of code in the MATLAB code.

The third field method of structure FDCPar stores in a string (between quotes) the name of the method that is used to minimize the SSE criterion of Equation (25). The user can select among the four different optimization methods discussed in section 2.6 of this paper. These methods are activated by setting field method equal to 'LM' for Levenberg-Marquardt, 'SP' for the downhill-Simplex method, and 'DE' and 'CMA' for global optimization with Differential Evolution or the Covariance Matrix Adaptation Evolutionary Strategy, respectively. The three different fields of structure FDCPar have to be defined by the user as they do not have default settings. An error will be printed to the screen if the user does not correctly specify the content of each field of FDCPar. This error gives detailed feedback to the user of what is wrong.

Once the model has been defined, the FDCFIT toolbox knows exactly which parameter space to search in pursuit of the global minimum. This is done by the optimization algorithm. The upper and lower bounds of the parameters of each model are hardwired in the MATLAB package and the global minimum is present within these ranges.

3.2. Input argument 2: E

The second input argument of the FDCFIT function, the variable E stores a $n \times 1$ vector with measured exceedance probabilities, $\hat{\mathbf{E}} = \{\hat{e}_1, \ldots, \hat{e}_n\}$. This vector is derived from the measured discharge record using the procedure detailed in section 2.1. This second input argument is also referred to as the empirical exceedance probability.

3.3. Input argument 3: Y

The third input argument of the FDCFIT function, the variable Y stores a $n \times 1$ vector with observed discharge values, $\hat{\mathbf{Y}} = \{\hat{y}_1, \ldots, \hat{y}_n\}$. Each entry of Y corresponds to their counterpart stored in the input argument E, and jointly they defined the measured (or empirical) FDC.

The number of elements of Y and E should match exactly, otherwise FDCFIT will produce an error and the code will terminate prematurely.

3.4. (Optional) input argument 4: `optim`

The third input argument of the main function FDCFIT is optional and determines the optimization algorithm and numerical settings that will be used for calibration of the FDC model parameters. Table 4 lists the different fields of `optim`.

Table 4: Content of the third (optional) input argument `optim` of the main program FDCFIT of the MATLAB toolbox. Each row signifies a different field of `FDCPar` and summarizes name, content, options, default settings, and variable type.

Field `optim`	Description	Options	Type	Default
TolX	Tolerance parameters	≥ 0	real	0.01
TolFun	Tolerance SSE	≥ 0	real	0.001
MaxFunEvals	Max model evaluations	≥ 100	integer	10,000
N	Max trials 'LM'/'SP'	≥ 1	integer	5
P	Population size 'DE'/'CMA'	≥ 1	integer	25
CR	Crossover value 'DE'	$(0, 1]$	real	0.8

The first three rows listed in Table 4 consider termination criteria of the optimization algorithm. The field TolX of input variable `optim` signifies the termination tolerance on the parameter values (positive scalar). The smaller the value of TolX the closer the final solution to the global minimum, yet at the expense of a larger number of model evaluations. The field TolFun defines the termination tolerance on the SSE objective function values (positive scalar). Small values of TolFun are necessary to converge in close vicinity of the global minimum. The field MaxFunEvals of structure `options` stores the maximum number of FDC model evaluations (positive integer) that the optimization algorithm is allowed to use. This computational budget cannot be exceeded - otherwise the optimizer will simply terminate its search and return the best parameter values thus far. The field MaxFunEvals thus provides a mechanism to escape from an unproductive parameter search.

The field N of structure `optim` lists the number of successive trials of the 'LM' or 'SP' optimization algorithms. The larger the value of field N the higher the likelihood that at least one of the trials will have converged successfully to the global minimum value of the SSE metric. A default value of $N = 5$ is deemed sufficient for most watersheds.

The field P denotes the population size that will be used by the DE and CMA algorithm. These two methods apply evolutionary principles to this ensemble of P individuals in pursuit of the global minimum of the SSE objective function. Finally, the field CR stores the crossover value that is used by the DE algorithm to create offspring from the parent population. The value of the crossover should be larger than zero and smaller than one. The lower the value of the crossover operator the "closer" the offspring population will be to the parent population, and thus the slower the rate of convergence to the global minimum. This slower rate of convergence has the desirable advantage that it gives opportunity to the algorithm to appropriate explore the parameter space. A crossover value of 0.8 (default) provides a good balance between exploration and exploitation and is deemed adequate for many parameter estimation problems. Note that the CMA algorithm does not use a crossover operator as it implements a sufficiently randomized parameter sampling strategy.

If the user does not specify the individual fields and/or their content of `optim`, then the FDCFIT toolbox will assume default settings. These values are listed in the last column of Table 4 and provide adequate

performance for the vast majority of the watersheds of the MOPEX data set.

3.5. (Optional) input argument 5: `options`

The fifth input argument of the function FDCFIT is the variable `options`. This input argument is optional, and is defined as structure array with the fields type and print (see Table 5.

Table 5: Content of the fifth (optional) input argument `options` of the main program FDCFIT of the MATLAB toolbox. Each row signifies a different field of `options` and summarizes name, content, options, default settings, and variable type.

Field `options`	Description	Options	Type	Default
type	Time-scale of flow duration curve	'daily'/'weekly'/'monthly'/'annual'	string	'daily'
print	Output printing to screen	'no'/'yes'	string	'yes'

The field type defines with a string enclosed between quotes the time scale of the discharge data and thus FDC. Examples include 'daily', 'weekly', 'monthly' or 'yearly'. This variable is only used for plotting of the results. The default setting of type = 'daily'. The field print of structure `options` controls the output writing of the FDCFIT toolbox. If print = 'yes' then the toolbox will print to the screen graphical output which summarizes in several tables and figures the results of the toolbox. The default setting of print = 'yes'.

The ascii-file `FDCFIT_output.txt` is printed to the MATLAB editor and lists the model and formulation that is being used, followed by a table with optimized values of the FDC model parameters and their corresponding RMSE value. The graphical output consists of two figures that compare the empirical (red dots) and the predicted (blue line) FDC using a linear (left) and logarithmic (right) y-scale for the streamflow values. This output is created by the function FDCFIT_plot (see Appendix B) which is called by the main program as follows

```
FDCFIT_plot(FDCPar,E,Y,x,RMSE,options);
```
(28)

The input arguments of this function have been defined by the user or are computed by the toolbox.

Appendix C presents the screen output that is being created for the first case study of section 4. Output writing is suppressed if field print of structure `options` is set to 'no'.

3.6. Output arguments

We now briefly discuss the two output (return) arguments of FDCFIT including `x`, and `RMSE`. These two variables summarize the results of the FDCFIT toolbox and are used for plotting of the results, and diagnostic analysis.

The variable `x` is a vector of size $1 \times d$ with the optimized values for the coefficients of the FDC model selected by the user. The root mean square error (RMSE) of the model fit (in streamflow or exceedance probability space) is stored as scalar in the variable `RMSE`.

4. Numerical examples

We now demonstrate the application of the FDCFIT package using daily discharge data from two basins in the United States, namely, the French Broad River basin at Asheville, North Carolina, and the Guadalupe River basin at Spring Branch, Texas. These are, respectively, the wettest and driest of the 12 MOPEX basins described in the study by *Duan et al.* (2006), and have been used by *Schoups and Vrugt* (2010) to introduce the generalized likelihood function (Bayesian inference).

4.1. Case Study I: French Broad River

The first case study involves daily streamflow observations (mm/day) from the French Broad river near Asheville, North Carolina. This data set (file '08167500.dly') is downloaded from the MOPEX ftp site `ftp://hydrology.nws.noaa.gov/pub/gcip/mopex/US_Data/` and stored in the folder "case_study_1" in the root directory of the FDCFIT toolbox. The following script is used in MATLAB to run the FDCFIT package.

MATLAB input script for case study I. The file "08167500.dly" contains the discharge data set of the French Broad river. A daily FDC is derived from this data set using the function `CalcFDC`. The variable `p_0` denotes the probability of zero rainfall days (or weeks/months/years, depending on the field **type** of structure `options`. The 2-parameter van Genuchten model (`FDCPar`.**model** = 8) of table 2 (`FDCPar`.**formulation** = 'E') is used to describe the empirical FDC. The Levenberg-Marquardt method (`FDCPar`.**method** = 'LM') is used to calibrate the model using a total of `optim`.**N** = 5 independent trials. A total of `optim`.**MaxFunEvals** = 10,000 function evaluations is allowed each trial and convergence is declared to the global optimum if subsequent changes in the parameter values or objective function values are smaller than `optim`.**TolX** = 1e-2 or `optim`.**TolFun** = 1e-3, respectively.

```
% ------------------------------------------------------------------- %
%                                                                     %
%  FFFFFFFFFFF DDDDDDDDD  CCCCCCCCCC FFFFFFFFFFF IIIIIIIIIIII TTTTTTTTTT  %
%  FFFFFFFFFFF DDDDDDDDDD CCCCCCCCC  FFFFFFFFFFF  IIIIIIIIII  TTTTTTTTTT  %
%  FF          DD      DD CC         FF               II          TT      %
%  FF          DD      DD CC         FF               II          TT      %
%  FFFFFF      DD      DD CC         FFFFFF           II          TT      %
%  FF          DD      DD CC         FF               II          TT      %
%  FF          DDDDDDDDDD CCCCCCCCC  FF           IIIIIIIIII      TT      %
%  FF          DDDDDDDDD  CCCCCCCCCC FF          IIIIIIIIIIII     TT      %
%                                                                     %
% ------------------------------------------------------------------- %

%% Which class of model and formulation? (defined by user)
FDCPar.model = 8;           % FDC model [1...15] ( see Table 1 and 2 )
FDCPar.formulation = 'E';   % Formulation 'Y' = f(E) or 'E' = f(Y)
FDCPar.method = 'LM';       % Optimization method: Levenberg-Marquardt

%% Define fields of structure optim
optim.TolX = 1e-2;          % Termination criteria parameters ( def: 1e-2 )
optim.TolFun = 1e-3;        % Termination criteria SSE value  ( def: 1e-3 )
optim.MaxFunEvals = 1e4;    % Maximum function evaluations    ( def: 1e4 )
optim.N = 5;                % Number of trials 'LM' or 'SP'   ( def: 5 )

%% Define fields of structure options
options.type = 'daily';     % Lets do daily flow duration curve
options.print = 'yes';      % Print output to screen

% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% %                          SETUP STUDY                              % %
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %

case_study = 1;                                     % Which case study?
addpath([pwd '\case_study_' num2str(case_study)]);  % Add to path

% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% %                            LOAD DATA                              % %
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %

ID = '08167500';                            % Which watershed (.dly file)
[E,Y,p_0] = CalcFDC(ID,options.type,6);     % Return empirical FDC

% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% %                       DERIVE FITTING COEFFICIENTS                 % %
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
[ x , RMSE ] = FDCFIT ( FDCPar , E , Y , optim , options );
```

From the content of the fields of structure `FDCPar` we can see that we are using the inverse formulation, $e = \mathcal{F}(y)$ of the 2-parameter van Genuchten model (model = 8 and formulation = 'E'). The mathematical expression of this model is listed under [8] in Table 2 and calibrated against the empirical FDC stored in structure `Meas_info` using N = 5 trials with the method = 'LM' (Levenberg-Marquardt) algorithm. Default values are used for the convergence criteria TolX, TolFun and MaxFunEvals.

Figure 3 plots the observed (red dots) and fitted (blue line) FDC using a (A) linear and (B) logarithmic scale of the streamflow values. This plot is automatically generated by the function FDCFIT_PLOT of the FDCFIT toolbox, details of which are provided in section .

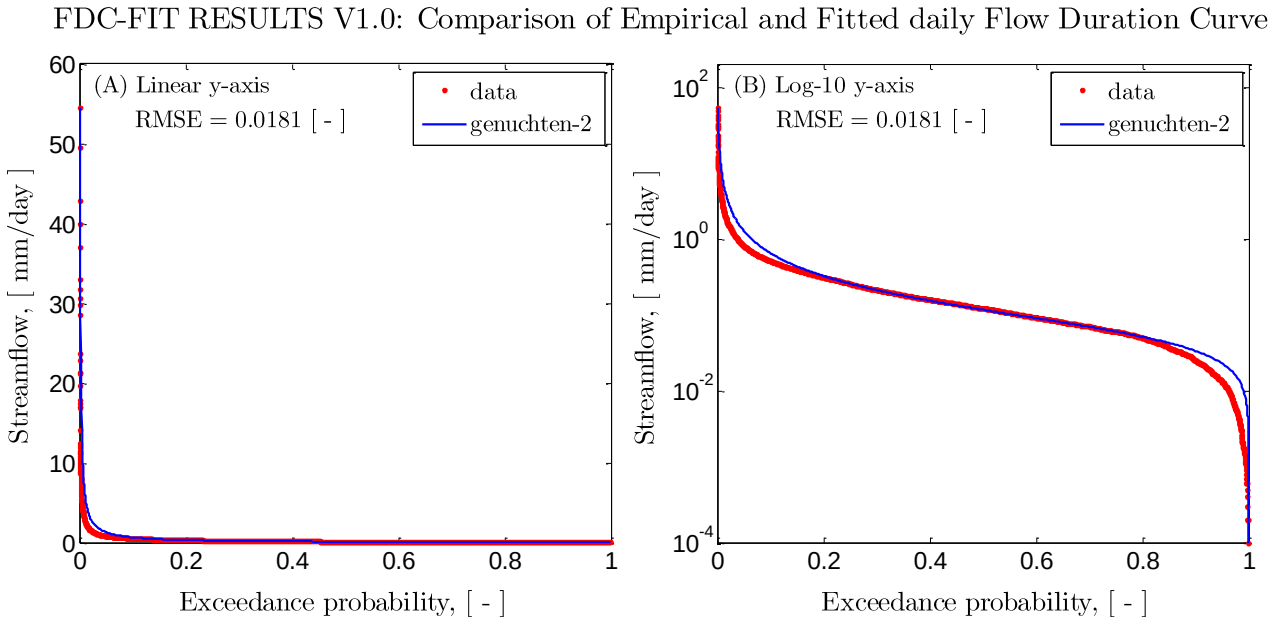


Figure 3: Comparison of the observed (red dots) and fitted (blue line) daily FDC using the 2-parameter van Genuchten model. The two plots differ in their y-scale (left: linear, right: logarithmic) to better visualize the results for the tails of the FDC.

The 2-parameter van Genuchten model mimics reasonably well the empirical FDC. Some deviations are visible in the tails of the FDC at low and high streamflow values respectively. A much improved fit to the empirical FDC is possible if the 3-parameter formulation of van Genuchten or Kosugi is used. We refer the reader to the work of *Sadegh et al.* (2016) for a comprehensive analysis of the newly proposed and existing FDC models for the MOPEX data set.

4.2. Case Study II: Gaudalupe River basin

The second case study involves daily streamflow data (mm/day) from the Guadalupe river basin at Spring Branch, Texas. The data file '03443000.dly' is downloaded from the MOPEX ftp site `ftp://hydrology.nws.noaa.gov/pub/gcip/mopex/US_Data/` and stored in the folder "case_study_2" in the root directory of the FDCFIT toolbox. The following input file is used for the FDCFIT toolbox.

MATLAB input script for case study II. The file "03443000.dly" contains the discharge data set of the Gaudalupe river basin in the USA. A daily FDC is derived from this data set using the function `CalcFDC`. The variable `p_0` denotes the probability of zero rainfall weeks (`options`.type = 'weekly'). The 3-parameter van Kosugi model (`FDCPar`.model = 15) of table 1 (`FDCPar`.formulation = 'Y') is used to describe the empirical FDC. The Levenberg-Marquardt method (`FDCPar`.method = 'LM') is used to calibrate the model using a total of `optim`.N = 5 independent trials. A total of `optim`.MaxFunEvals = 10,000 function evaluations is allowed each trial and convergence is declared to the global optimum if subsequent changes in the parameter values or objective function values are smaller than `optim`.TolX = 1e-2 or `optim`.TolFun = 1e-3, respectively.

```
% ------------------------------------------------------------------- %
%                                                                     %
%  FFFFFFFFFFF DDDDDDDDD  CCCCCCCCCC FFFFFFFFFFF IIIIIIIIIIII TTTTTTTTTT  %
%  FFFFFFFFFFF DDDDDDDDDD CCCCCCCCC  FFFFFFFFFFF  IIIIIIIIII  TTTTTTTTTT  %
%  FF          DD      DD CC         FF               II          TT      %
%  FF          DD      DD CC         FF               II          TT      %
%  FFFFFF      DD      DD CC         FFFFFF           II          TT      %
%  FF          DD      DD CC         FF               II          TT      %
%  FF          DDDDDDDDDD CCCCCCCCC  FF           IIIIIIIIII      TT      %
%  FF          DDDDDDDDD  CCCCCCCCCC FF          IIIIIIIIIIII     TT      %
%                                                                     %
% ------------------------------------------------------------------- %

%% Which class of model and formulation? (defined by user)
FDCPar.model = 15;           % FDC model [1...15] ( see Table 1 and 2 )
FDCPar.formulation = 'Y';    % Formulation 'Y' = f(E) or 'E' = f(Y)
FDCPar.method = 'LM';        % Optimization method: Levenberg-Marquardt

%% Define fields of structure optim
optim.TolX = 1e-2;           % Termination criteria parameters ( def: 1e-2 )
optim.TolFun = 1e-3;         % Termination criteria SSE value  ( def: 1e-3 )
optim.MaxFunEvals = 1e4;     % Maximum function evaluations    ( def: 1e4 )
optim.N = 5;                 % Number of trials 'LM' or 'SP'   ( def: 5 )

%% Define fields of structure options
options.type = 'daily';      % Lets do daily flow duration curve
options.print = 'yes';       % Print output to screen

% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% %                          SETUP STUDY                              % %
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %

case_study = 2;                                    % Which case study?
addpath([pwd '\case_study_' num2str(case_study)]);  % Add to path

% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% %                            LOAD DATA                              % %
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %

ID = '03443000';                            % Which watershed (.dly file)
[E,Y,p_0] = CalcFDC(ID,options.type,6);     % Return empirical FDC

% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
% %                       DERIVE FITTING COEFFICIENTS                 % %
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
[ x , RMSE ] = FDCFIT ( FDCPar , E , Y , optim , options );
```

The field model of structure `FDCPar` is set equal to 15 which means that the 3-parameter formulation of the Kosugi model is used to mimic the empirical weekly FDC. We use the forward formulation (formulation = 'Y') listed in Table 1 of this manual. The downhill Simplex algorithm, (`options`.method = 'SP') is used with N = 5 different trials.

Figure 4 summarizes the results of the 3-parameter Kosugi model.

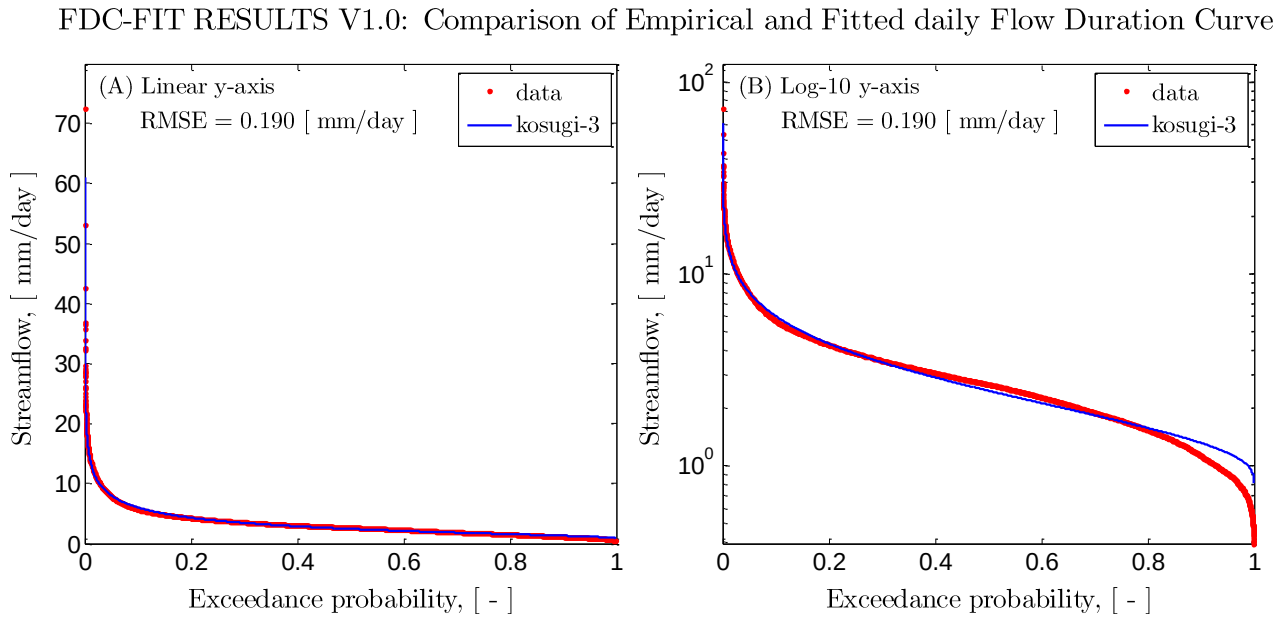


Figure 4: Comparison of the observed (red dots) and fitted (blue line) weekly FDC using the 3-parameter Kosugi expression. The plot on the left-hand side uses a linear scale whereas a logarithmic y-scale is used at the right-hand side to better visualize the results in the tails of the FDC.

The 3-parameter formulation of Kosugi matches closely a large portion of the FDC, yet the fit at the highest exceedance probabilities is somewhat poorly. This result is not surprising as the forward formulation of the model (see Table 1) necessitates calibration against the observed discharge data. This emphasizes fitting of the peak flows at low exceedance probabilities. The use of the ℓ_1-objective function in Equation (26) would resolve somewhat this problem as the errors in the peak flows would no longer dominate the parameter estimation. Alternatively, we can choose the inverse formulation of the 3-parameter Kosugi model, formulation = 'E', As this model requires fitting against the exceedance probabilities, this places equal weight on each portion of the FDC.

Table 6 summarizes the quality of fit (RMSE) for each of the models of the MATLAB toolbox and their forward and inverse formulations using the empirical weekly FDC of the French Broad watershed.

Table 6: Fitting results for the fifteen different models listed in Tables 1 (formulation = 'Y') and 2 (formulation = 'E') for the empirical daily FDC of the French Broad River basin near Asheville, NC. The listed values in the third column signify the RMSE between the empirical and predicted FDC.

Model name	model	formulation†	
		$y = \mathcal{F}(e)$	$e = \mathcal{F}^{-1}(y)$
Lognormal	[1]	0.2650	0.0099
Gumbel	[2]	0.9855	0.0472
Logistic	[3]	1.0377	0.0662
Logarithmic	[4]	0.9374	0.0207
Power	[5]	0.4238	0.1817
Quimpo	[6]	0.4982	0.1099
Viola	[7]	0.3917	0.0062
van Genuchten	[8]	0.2629	0.0181
Kosugi	[9]	0.2650	0.0099
Lognormal	[10]	0.2627	0.0079
Generalized Pareto	[11]	0.3451	0.0138
Extreme Value	[12]	0.3562	0.0041
Franchini and Suppo	[13]	0.4406	0.0948
van Genuchten	[14]	0.2564	0.0058
Kosugi	[15]	0.2627	0.0079

†: Units of third and fourth column are (mm/day) and (-) respectively

The proposed parametric expression (van Genuchten and Kosugi) exhibit an excellent performance. Their RMSE values are substantially lower than most of the literature FDC models. This most evident when the forward formulation of each model is used, $y = \mathcal{F}(e)$ which necessitates calibration against the measured streamflow values. The two and three-parameter lognormal models exhibit an almost similar performance for this particular watershed. The Franchini and Suppo and Quimpo formulations are particularly deficient - and unable to closely mimic the empirical weekly FDC. *Sadegh et al.* (2016) evaluates the different parametric expressions of Tables 1 and 2 for a large suite of watersheds of the MOPEX data set. Readers are referred to this publication for further details about the performance and merits of each individual FDC model. This paper also evaluates the regionalization potential of each model by correlating their parameters against a large number of catchment characteristics. The proposed FDC models are of great value for diagnostic model evaluation as their parameters can act as summary statistics (*Vrugt and Sadegh*, 2013). What is more, temporal analysis of the parameter values of the proposed models can help detect and diagnose catchment nonstationarity (*Sadegh et al.*, 2015).

5. Scaling of flow duration curves

The use of equation 19 enables the use of physically based scaling (*Tuli et al.*, 2001) to coalesce the FDCs into a single reference curve using scaling factors that describe the set as a whole. This opens up new ways for catchment classification, geostatistical analysis and regionalization.

Figure 5 compares the original (unscaled) and scaled FDCs of the MOPEX data set derived by application of the scaling theory of *Tuli et al.* (2001). The solid line represents the reference curve.

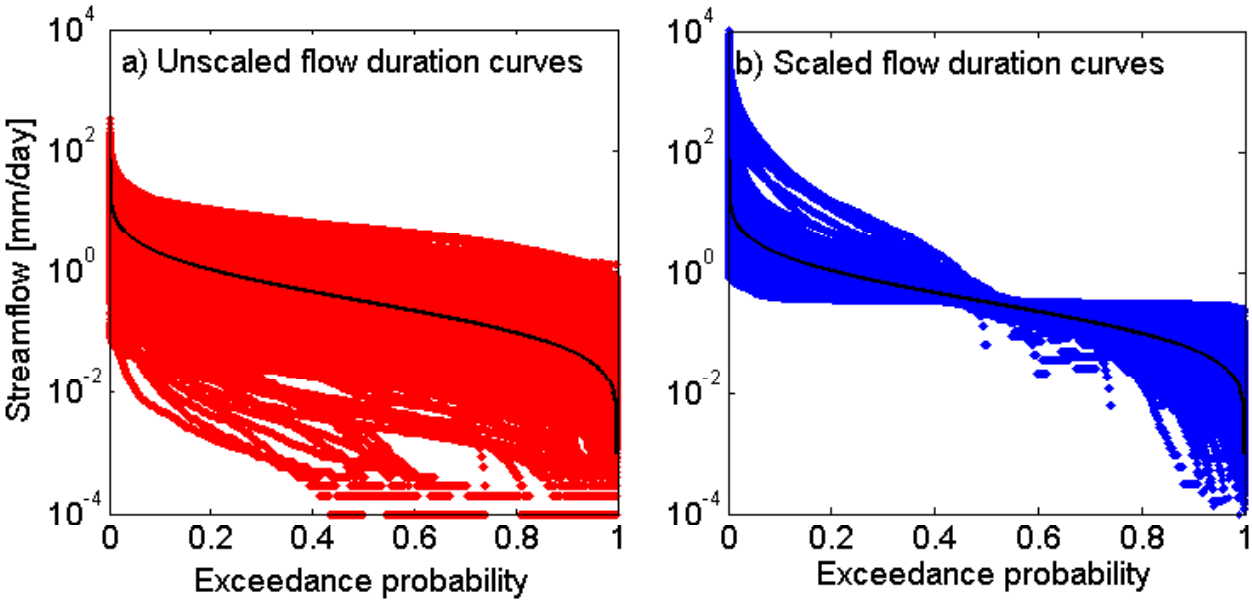


Figure 5: Application of physically-based scaling to flow duration curves of MOPEX data set: a) unscaled data, b) scaled data.

The scaled data groups much closer around the reference curve - which demonstrates that we can define with a single scaling factor each observed FDC. A correlation coefficient of 0.87 is found (not shown) between scaling factors and basic watershed properties. This provides new opportunities for regionalization. The FDC reference curves of different continents (countries) can serve as benchmark for prediction in ungaged basins. A publication on scaling and regionalization of FDCs is forthcoming (*Naeini and Vrugt*, 2016).

6. Summary

In this paper we have introduced a MATLAB package, entitled FDCFIT, which provides hydrologist with a new class of parametric functions of the flow duration curve. The coefficients (parameters) in these expressions are fitted automatically using data from an empirical FDC. Graphical output is provided as well. Two different case studies were used to illustrate the main capabilities and functionalities of the MATLAB toolbox. These example studies are easy to run and adapt and serve as templates for other modeling problems and watershed data sets.

The toolbox allows for determination of the daily, weekly, monthly and annual FDC - yet in our work we have not analyzed the relationship between these different curves and their optimized parameter values. Also, the parametric expressions of the FDC used herein apply directly to fitting of the annual peak flow curve as well - a powerful alternative to the log-Pearson type-III distribution advocated by the USGS in their 1982 contribution (*IACWD*, 1982) and used worldwide by many researchers to model flood flow frequencies. Much additional work is required to adopt this new methodology - with the advantage that it is easy to implement and provides estimates of flood-flow frequency estimates as key element to flood damage control.

7. Acknowledgments

The MATLAB toolbox of FDCFIT is available upon request from the first author, `jasper@uci.du`.

Appendix A. Download and installation

The FDCFIT code can be downloaded from my website at the following link `http://faculty.sites.uci.edu/FDCFIT`. Please save this file called "MATLAB-pCode-FDCFIT-V1.0" to your hard disk, for instance, in the directory "D:\Downloads\Toolboxes \MATLAB\FDCFIT". Now open Windows explorer in this directory (see Figure A.1).

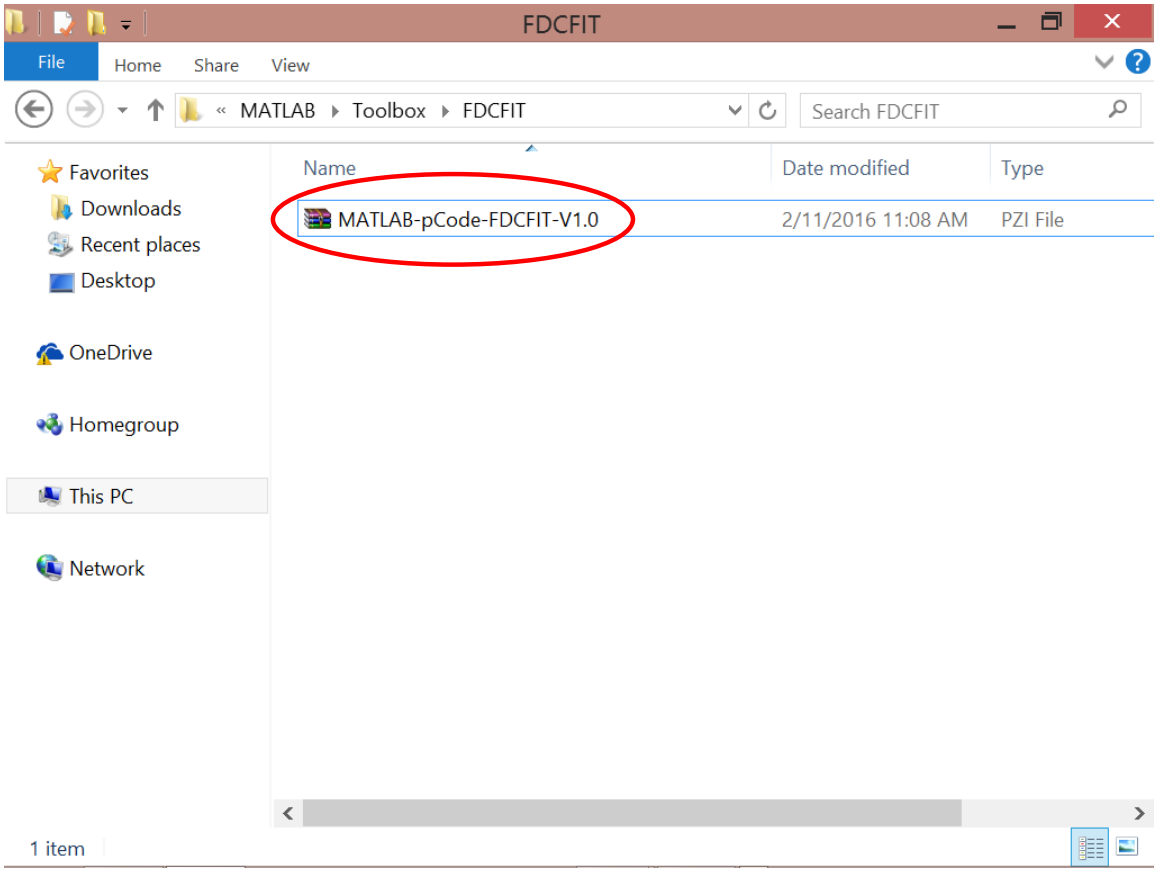


Figure A.1

You will notice that the file does not have an extension - it is just called **MATLAB-pCode-FDCFIT-V1.0**. That is because Windows typically hides extension names.

If you can already see file extensions on your computer, then please skip the next step. If you cannot see the file extension, please click the **View** tab. Then check the box titled "File name extensions" (see Figure A.2).

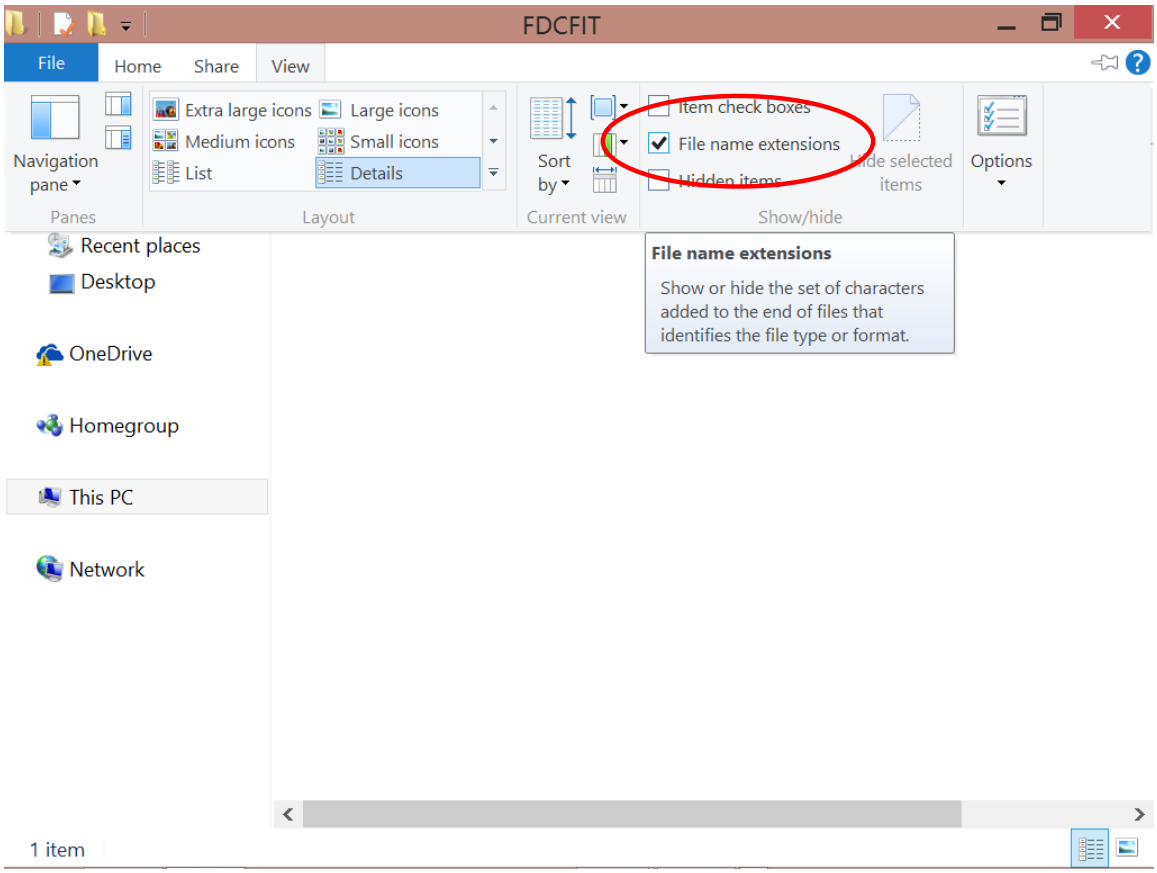


Figure A.2

Now you should be able to see the file extension. Right-click the file name and select **Rename** (see Figure A.3).

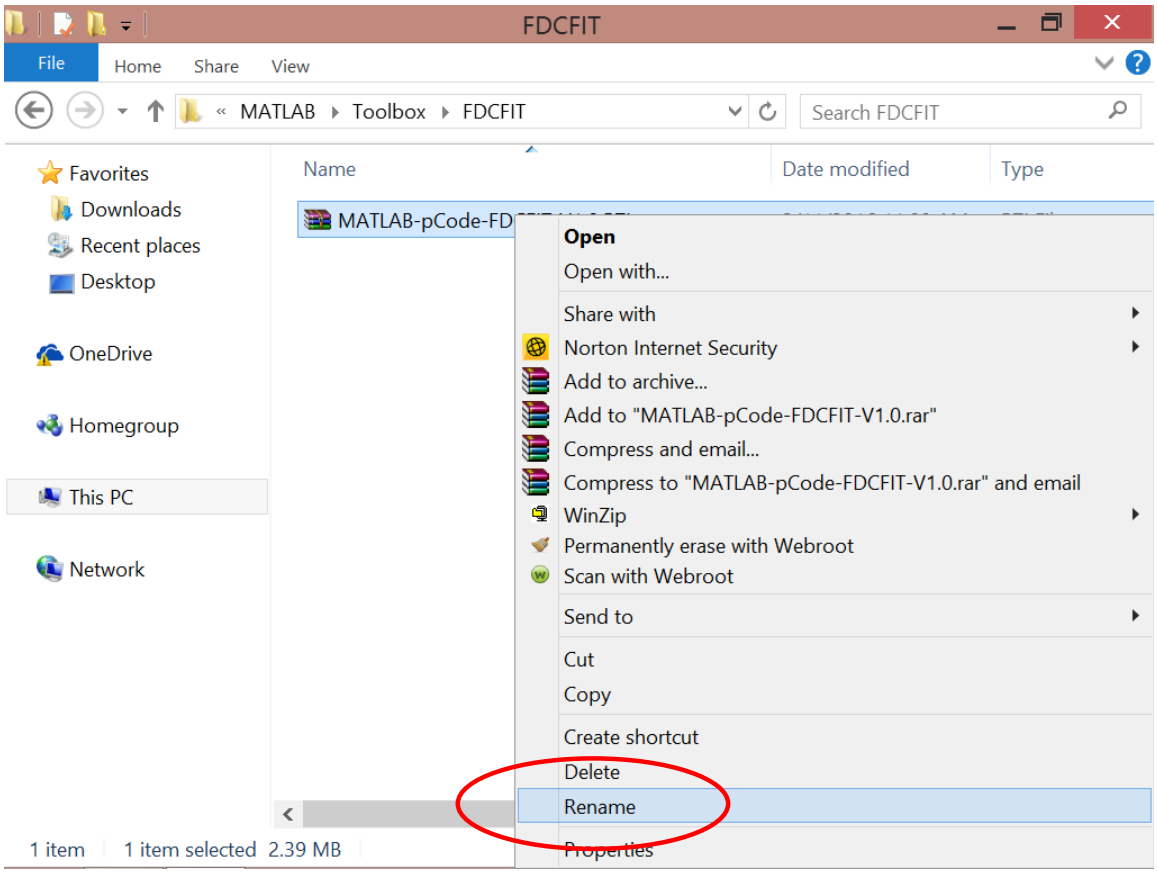


Figure A.3

Now change the extension of "MATLAB-pCode-FDCFIT-V1.0" from ".pdf" to ".rar" (see Figure A.4).

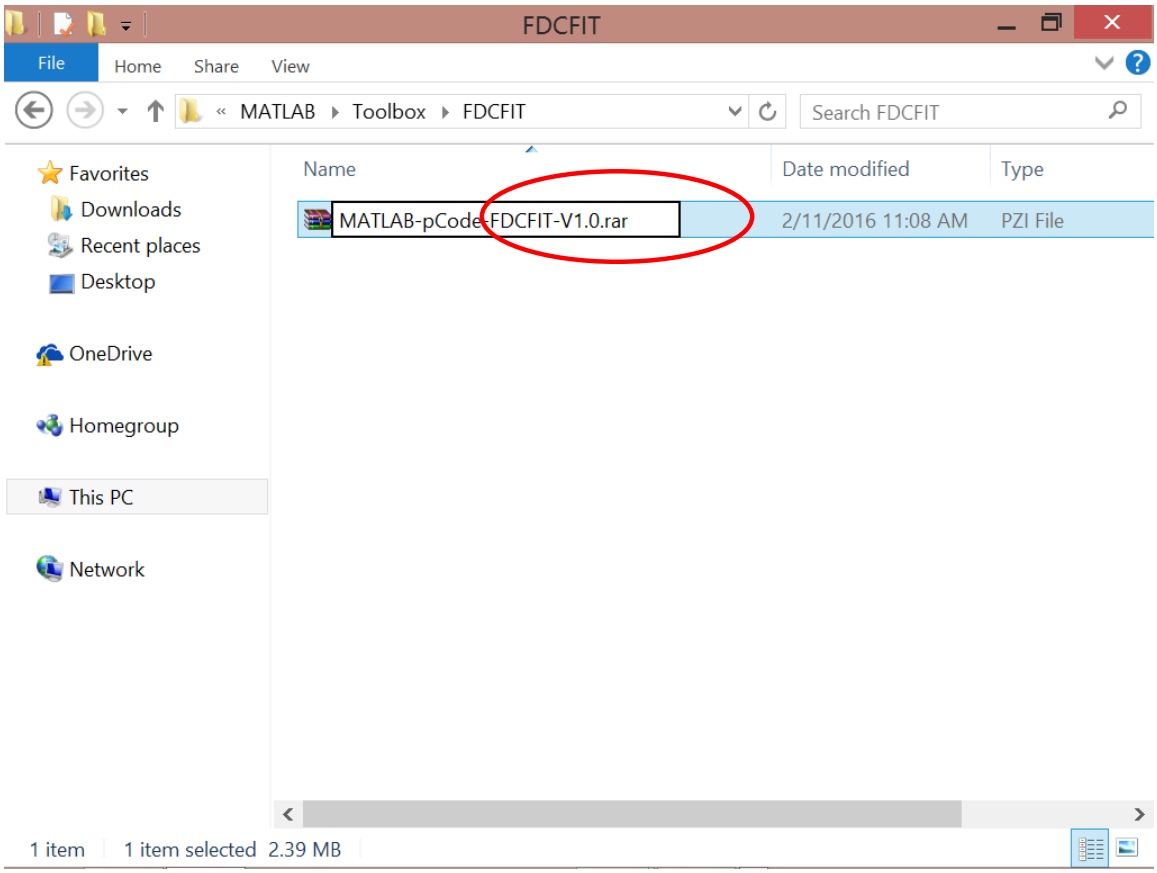


Figure A.4

After entering the new extension, hit the **Enter** (return) key. Windows will give you a warning that the file may not work properly (see Figure A.5). This is quite safe - remember that you can restore the original extension if anything goes wrong.

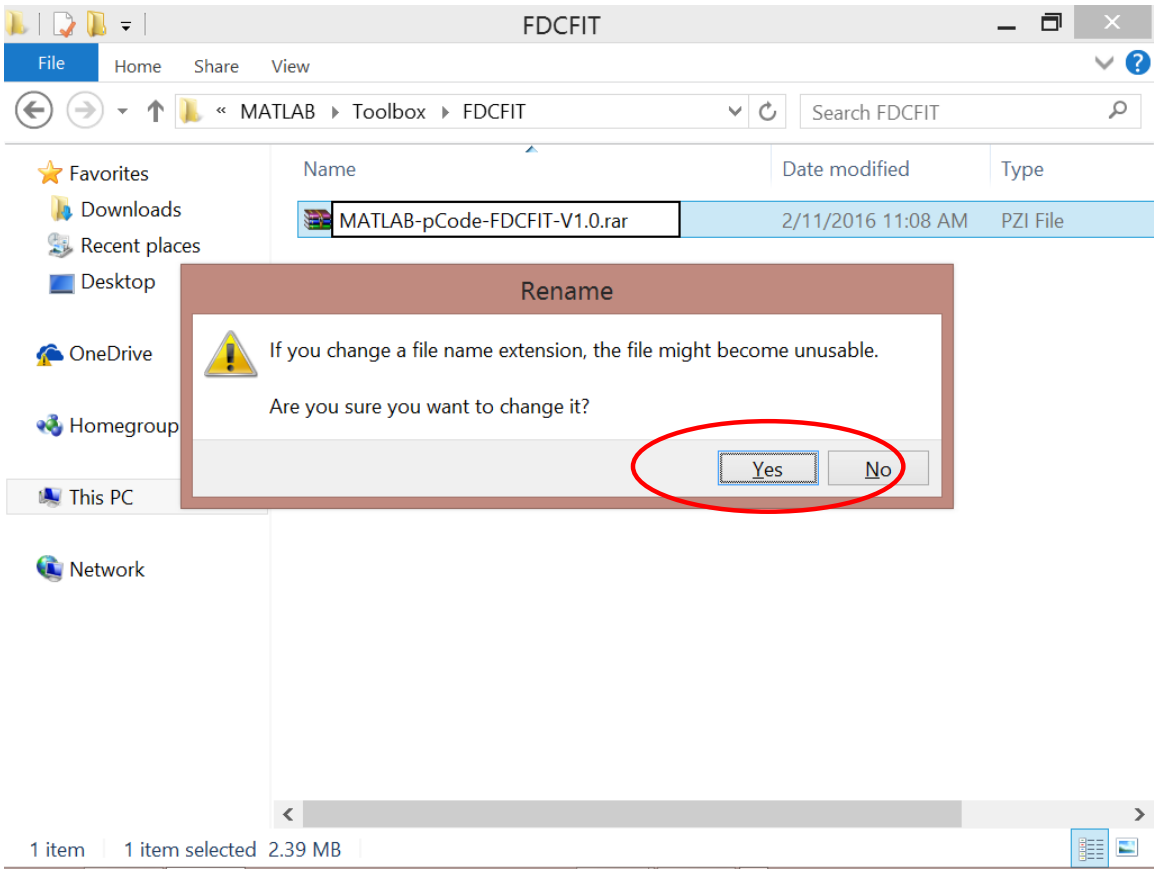


Figure A.5

It is also possible that you might get another message telling you that the file is "read-only". In this case either say yes to turning off read-only, or right-click the file, select **Properties** and uncheck the **Read-only** box.

If you do not have permission to change the file extension, you may have to login as Administrator. Another option is to make a copy of the file, rename the copy and then delete the original.

Now you have changed the extension of the file to ".rar" you can use the program WinRAR to extract the files to whatever folder your desire, for instance "D:\Downloads\Toolboxes\MATLAB\FDCFIT". Right-click the file name and select **Extract Here** (see Figure A6).

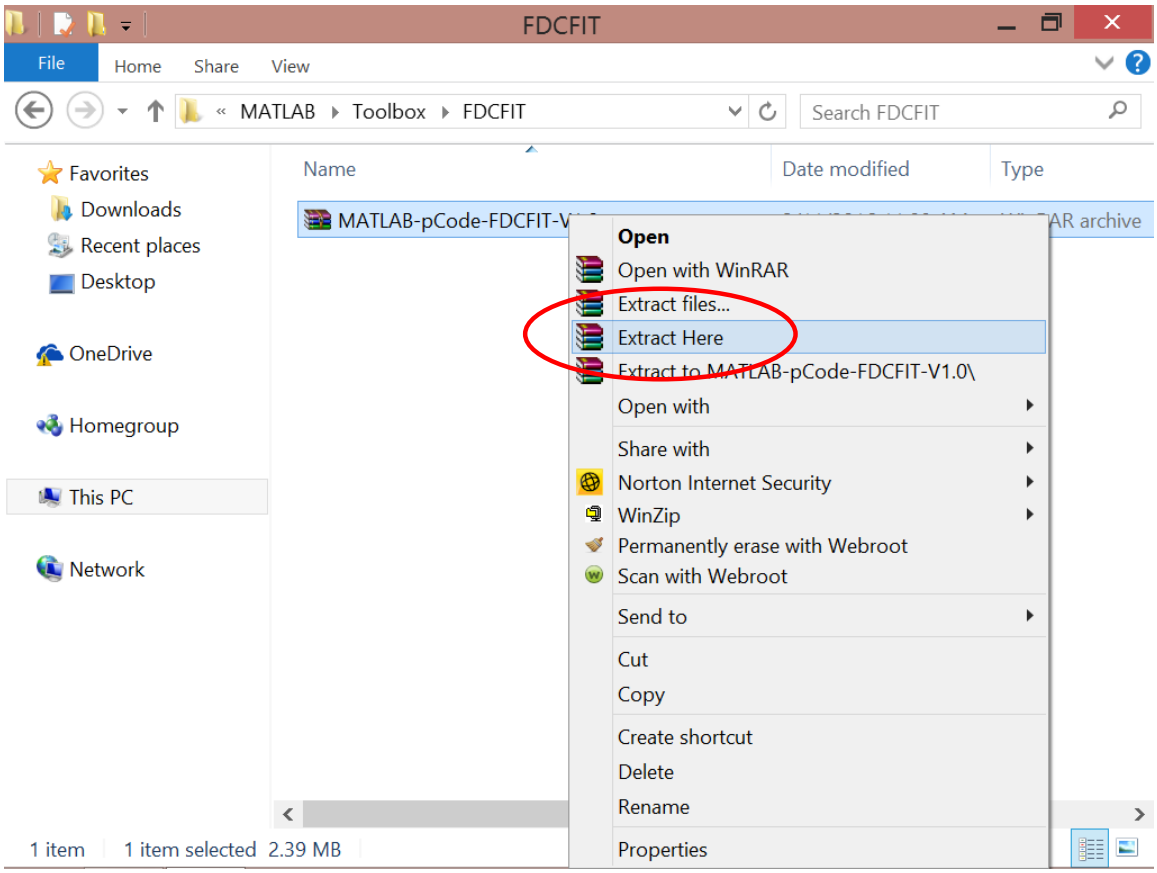


Figure A.6

Now WinRAR should extract the files to your folder. The end result should look as in Figure A.7.

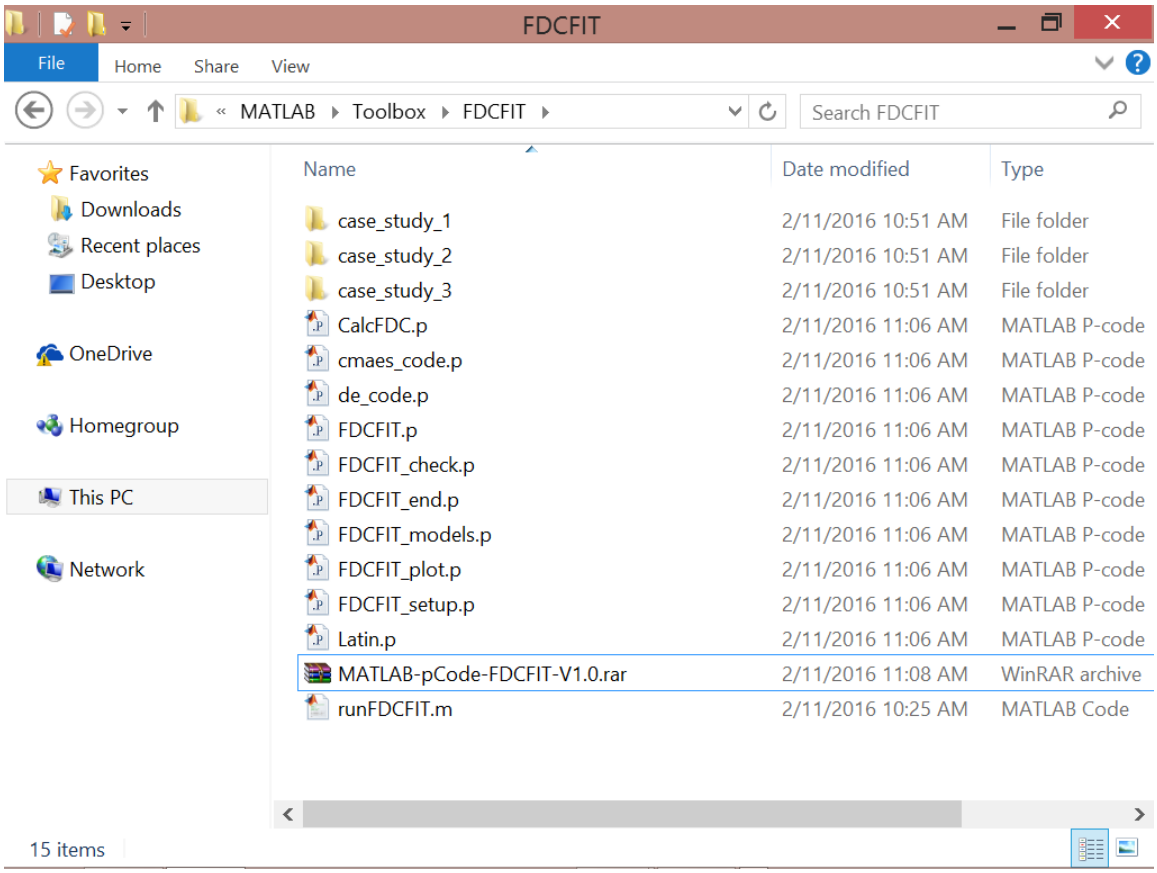


Figure A.7

The FDCFIT toolbox is now ready for use in MATLAB.

Appendix B. Main functions of the FDCFIT toolbox

Table B.1 summarizes, in alphabetic order, the different function/program files of the FDCFIT toolbox in MATLAB. The main program RUNFDCFIT contains three prototype studies which involve the discharge response of the driest and wettest watershed on record of the original MOPEX data set. These example studies provide a template for users to setup their own case study. The last line of each example study involves a function call to FDCFIT, which uses all the other functions listed above to derive the fitting coefficients of the parametric expressions selected by the user. Each case study of RUNFDCFIT uses its own directory which stores the necessary data required by the program.

The function FDCFIT_PLOT visualizes the results (output arguments) of FDCFIT. This includes a Table named "FCFIT_output.txt" with the fitting results (optimized parameter values and fitting statistics) written to the screen in the MATLAB editor and two figures (linear and log-space) with a comparison of the observed (empirical) and predicted FDC. The user can control output writing is output will be written to the screen if the field print of structure `options` is set to 'yes'. This constitutes the default setting. Output writing can be suppressed by setting print = 'no'. Appendix C presents the graphical output of the FDCFIT toolbox for the first case study of section 4.

Those users that do not have access to the optimization toolbox need an alternative search algorithm to determine the values of the fitting coefficients of each FDC model. Those users that do not have a license of the optimization toolbox are required to either use the DE or CMA global search algorithms for fitting of the FDC model parameters. Their scripts are provided along with the other functions of the toolbox. Users can also combine this toolbox with the DREAM algorithm (*Vrugt*, 2016) - something that is straightforward to do and as byproduct also provides estimates of the posterior parameter and model uncertainty. Knowledge of these uncertainties is key for diagnostic model evaluation - in which the parameters are used as summary metrics.

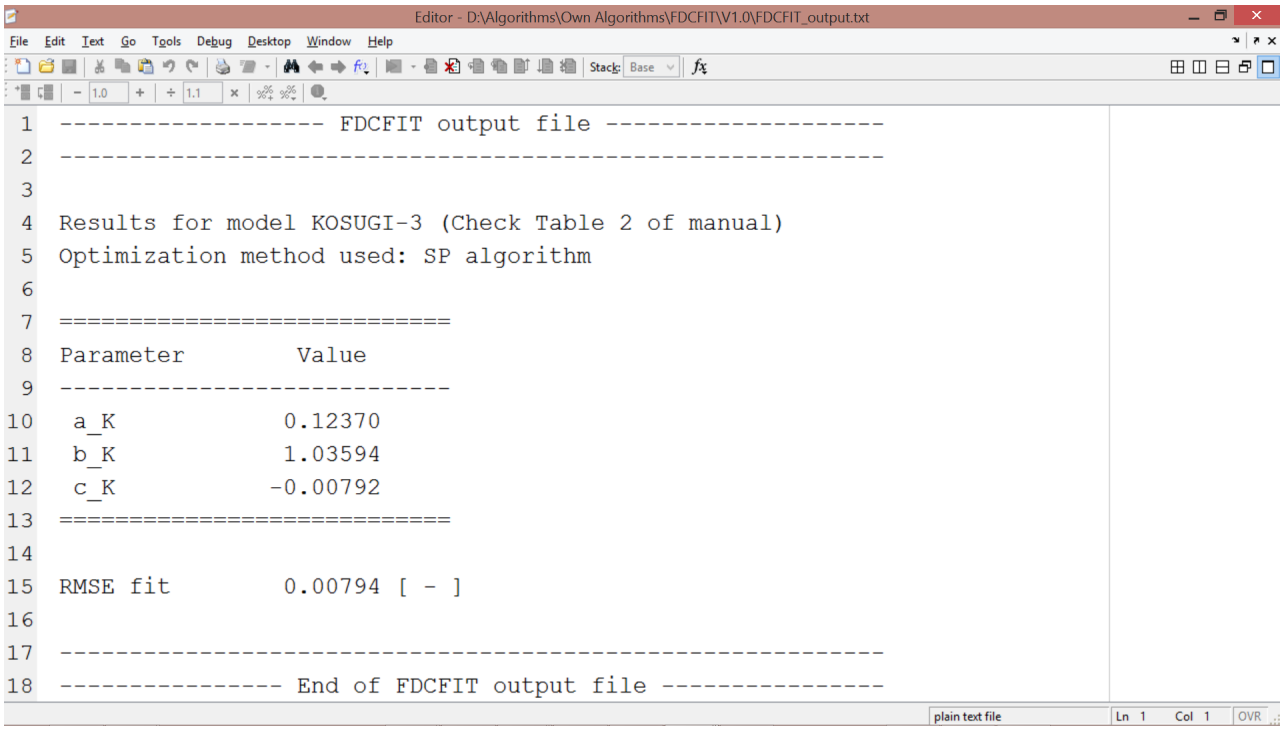
Table B.1: Description of the MATLAB functions and scripts (.m files) used by FDCFIT, version 1.0.

Function name	Description
CalcFDC	Calculates the daily/weekly/monthly/yearly flow duration curve from a time series of discharge data
FDCFIT	Calls different functions and derives the fitting coefficients of a given parametric expression
FDCFIT_check	Checks the settings supplied by the user and if inconsistent returns a warning
FDCFIT_end	Isolates return arguments from data and prints to screen results of fitting
FDCFIT_plot	Graphical output - plots the fitted flow duration curve in linear and log-scale and compares with data
FDCFIT_setup	Defines the ranges of the coefficients in each of the parametric expressions
FDCFIT_models	Returns the simulated exceedance probabilities of FDC model selected by user
Latin	Latin hypercube sampling used to initialize trajectories for multistart local search methods
cmaes_code	Implementation of the covariance matrix adaptation evolutionary strategy
de_code	Implementation of the differential evolution global optimization algorithm
fminsearch	Built-in MATLAB function for multidimensional unconstrained nonlinear minimization (Nelder-Mead)
lsqnonlin	Built-in MATLAB function that solves nonlinear least squares optimization problems (Levenberg-Marquardt)
runFDCFIT	Main script of toolbox - setup of problem and call of main function

Appendix C. Screen output

The MODELAVG toolbox presented herein returns to the user tables and figures which jointly summarize the results of the toolbox. This appendix displays all this output for the first case study involving the FDC of the French Broad river watershed in the USA. We use the three parameter Kosugi model, K-3, to fit the empirical FDC.

The file "FDCFIT_output.txt" summarizes the setup and results of FDCFIT. This includes a Table with the optimized values of the parameters and the corresponding RMSE values. Figure C1 displays a screen shot of the MATLAB command window after the program FDCFIT has terminated its calculations.




```
------------------- FDCFIT output file --------------------
-----------------------------------------------------------

Results for model KOSUGI-3 (Check Table 2 of manual)
Optimization method used: SP algorithm

============================
Parameter        Value
----------------------------
 a_K            0.12370
 b_K            1.03594
 c_K           -0.00792
============================

RMSE fit        0.00794 [ - ]

-----------------------------------------------------------
---------------- End of FDCFIT output file ----------------
```

Figure C1: Screen print of ascii file "FDCFIT_output.txt" after FDCFIT has terminated case study 2. The optimized values of the coefficients are listed using a notation that matches exactly the Equations presented herein in section 2.3, and 2.4. For convenience, the print out also displays the RMSE of the least squares fit.

The toolbox also presents to the user two figures that visualize the results (see Figure C2) Note that the legend and RMSE values in both graphs adapt automatically to the model and formulation being used.

FDC-FIT RESULTS V1.0: Comparison of Empirical and Fitted daily Flow Duration Curve

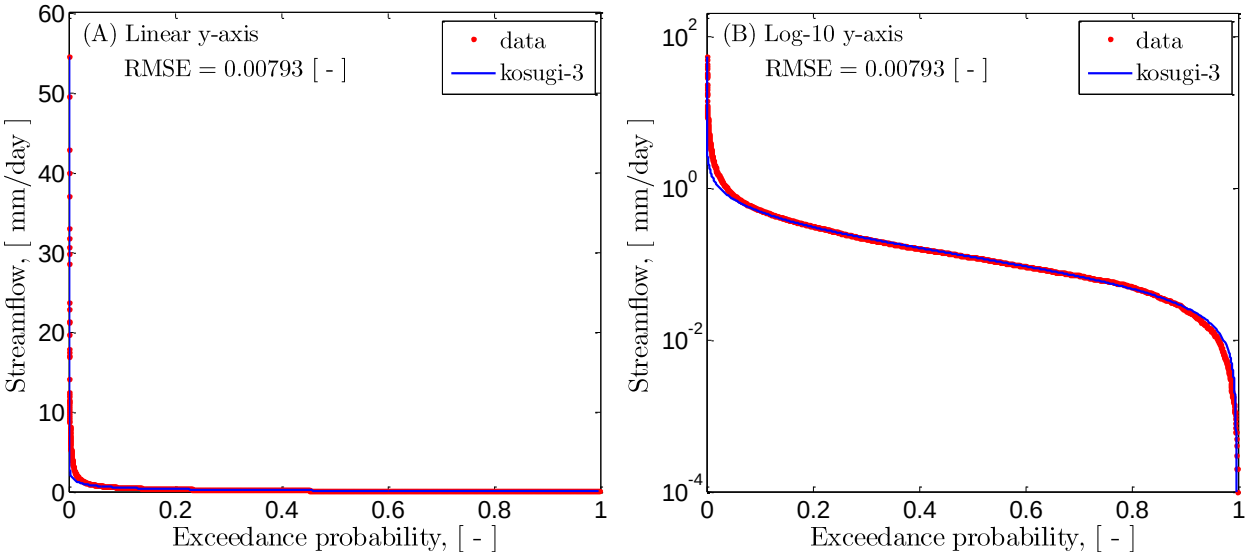


Figure C2

Notation (symbol use) is consistent with the different parametric expressions listed in section 2.3, and summarized in Tables 1 and 2. Once the parameters have been determined, they can (among others) be used within their respective parametric expression to predict the streamflow for a given exceedance probability (formulation = 'Y' in Table 1) or vice-versa the exceedance probability as function of the streamflow (formulation = 'E' in Table 2). The optimized parameter values can also be used for scaling and geostatistical analysis.

Appendix D. References

D.J. Booker, and T.H. Snelder, "Comparing methods for estimating flow duration curves at ungauged sites," *Journal of Hydrology*, vol. 434-435, pp. 78-94, 2012.

Q. Duan, J. Schaake, V. Andréassian, S. Franks, G. Goteti, H.V. Gupta, Y.M. Gusev, F. Habets, A. Hall, L. Hay, T. Hogue, M. Huang, G. Leavesley, X. Liang, O.N. Nasonova, J. Noilhan, L. Oudin, S. Sorooshian, T. Wagener, E.F. Wood, "Model Parameter Estimation Experiment (MOPEX): An overview of science strategy and major results from the second and third workshops," *Journal of Hydrology*, vol. 320, pp. 3-17, doi:10.1016/j.jhydrol.2005.07.031, 2006.

N. Fennessey, and R.M. Vogel, "Regional flow-duration curves for ungauged sites in Massachusetts, *Journal of Water Resources Planning and Management*, vol. 116, pp. 530-549, 1990.

H.A. Foster, "Duration curves," *ASCE Transactions*, vol. 99, pp. 1213-1267, 1934.

M. Franchini, and M. Suppo, "Regional analysis of flow duration curves for a limestone region," *Water Resources Management*, vol. 10, pp. 199-218, 1996.

N. Hansen, and A. Ostermeier, "Completely derandomized self-adaptation in evolution strategies", *Evolutionary Computation*, vol. 9 (2), pp. 159-195, 2001.

R.W. Katz, M.B. Parlange, and P. Naveau, "Statistics of extremes in hydrology," *Advances in Water Resources*, vol. 25, no. 8-12, pp. 1287-1304, 2002.

K. Kosugi, "Three-parameter lognormal distribution model for soil water retention," *Water Resources Research*, vol. 30, no. 4, pp. 891-901, 1994.

K. Kosugi, "Lognormal distribution model for unsaturated soil hydraulic properties," *Water Resources Research*, vol. 32, no. 9, pp. 2697-2703, 1996.

Longobardi, A., and P. Villani, "A statistical, parsimonious, empirical framework for regional flow duration curve shape prediction in high permeability Mediterranean region," *Journal of Hydrology*, 507, 174-185, 2013.

D.W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *Journal of the Society for Industrial & Applied Mathematics*, vol. 11, no. 2, pp. 431-441, 1963.

M. Naeini, and J.A. Vrugt, "Miller similarity and scaling of flow duration curves: Theory, numerical implementation and regionalization," *Water Resources Research*, vol. XX, pp. XX–XX, doi:10.1002/wrcr.XXXX, 2016.

J.A. Nelder, R. Mead, "A simplex method for function minimization," *The Computer Journal*, vol. 7, no. 4, pp. 308-313, doi:10.1093/comjnl/7.4.308, 1965.

R.G. Quimpo, A.A. Alejandrino, and T.A. McNally, "Regionalised flow duration curves for Philippines," *Journal of Water Resources Planning and Management*, vol. 109, no. 4, pp. 320-330, 1983.

M. Sadegh, and J.A. Vrugt (2013), "Approximate Bayesian Computation in hydrologic modeling: equifinality of formal and informal approaches," *Hydrology and Earth System Sciences*, vol. 17, 4831-4850, doi:10.5194/hess-17-4831-2013, 2013.

M. Sadegh, and J.A. Vrugt, "Approximate Bayesian Computation using Markov Chain Monte Carlo simulation: $\mathrm{DREAM}_{(\mathrm{ABC})}$", *Water Resources Research*, vol. 50, pp. 6767-6787, doi:10.1002/2014WR015386, 2014.

M. Sadegh, J.A. Vrugt, C. Xu, and E. Volpi, "The stationarity paradigm revisited: Hypothesis testing using diagnostics, summary metrics, and $\mathrm{DREAM}_{(\mathrm{ABC})}$", *Water Resources Research*, 51 (11), pp. 9207-9231, doi:10.1002/2014WR016805, 2015.

M. Sadegh, J.A. Vrugt, and H.V. Gupta, "The soil water characteristic as new class of closed-form parametric expressions for the flow duration curve," *Journal of Hydrology*, vol. 535, no. XX, pp. 438-456, doi:10.1016/j.jhydrol.2016.01.027, 2016.

E. Sauquet, and C. Catalogne, "Comparison of catchment grouping methods for flow duration curve estimation at ungauged sites in France, *Hydrology and Earth System Sciences*, vol. 15, pp. 2421-2435, 2011.

G. Schoups, and J.A. Vrugt, "A formal likelihood function for parameter and predictive inference of hydrologic models with correlated, heteroscedastic and non-gaussian errors," *Water Resources Research*, vol. 46, W10531, doi:10.1029/2009WR008933, 2010.

J.K. Searcy, "Flow-duration curves," *Geological water supply paper: Methods and practices of the Geological Survey*, pp. 1-33, 1959.

R. Storn, and K. Price, "Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, pp. 341-359, 1997.

A. Tuli, K. Kosugi, and J.W. Hopmans, "Simultaneouls scaling of soil water retention and unsaturated hydraulic conductivity functions assuming lognormal pore-size distribution," *Advances in Water Resources*, vol. 24, pp. 677-688, 2001.

M.T van Genuchten, "A closed-form equation for predicting the hydraulic conductivity of unsaturated soils," *Soil Science Society of America Journal*, vol. 44, no. 5, pp. 892-898, 1980.

F. Viola, L.V. Noto, M. Cannarozzo, and G. La Loggia, " Regional flow duration curves for ungauged sites in Sicily, *Hydrology and Earth System Sciences*, 15, pp. 323-331, 2011.

R.M. Vogel, and N.M. Fennessey, "Flow-duration curves I: New interpretation and confidence intervals, *Journal of Water Resources Planning and Management*, vol. 120, no. 4, pp. 485-504, 1994.

R.M. Vogel, and N.M. Fennessey, "Flow-duration curves II: A Review of applications in water resources planning, *Water Resources Bulletin*, vol. 31, no. 6, pp. 1029-1039, 1995.

J.A. Vrugt, and M. Sadegh, "Toward diagnostic model calibration and evaluation: Approximate Bayesian computation," *Water Resources Research*, vol. 49, doi:10.1002/wrcr.20354, 2013.

J.A. Vrugt, "Multi-criteria Optimization using the AMALGAM software package: Theory, concepts, and MATLAB Implementation, "*Manual, Version 1.0*, pp. 1-53, 2015a.

J.A. Vrugt, "MODELAVG: A MATLAB toolbox for postprocessing of model ensembles, "*Manual, Version 1.0*, pp. 1-44, 2015c.

J.A. Vrugt, "Markov chain Monte Carlo simulation using the DREAM software package: Theory, concepts, and MATLAB Implementation," *Environmental Modeling & Software*, vol. 75, pp. 273-316, 10.1016/j.envsoft.2015.08.013, 2016.

Interagency advisory commitee on water data, "Guidelines for determining flood flow frequency", Bulletin #17B of the hydrology subcommittee, US Department of the Interior, Geological Survey, Reston, VA 22092, 1982.