Study of recharging stations placement for electric car sharing systems

Report 3

Previous results

Car2go:

- Median parking duration per day per car = 820 min
- Median booking freq. per day per car = 5.95

Enjoy

- Median parking duration per day per car = 850 min
- Median parking freq. per day per car = 7.17

Those results assert that the cars are 1**4 hours per day parked**, thus 10h of bookings per day. This is not compatible with:

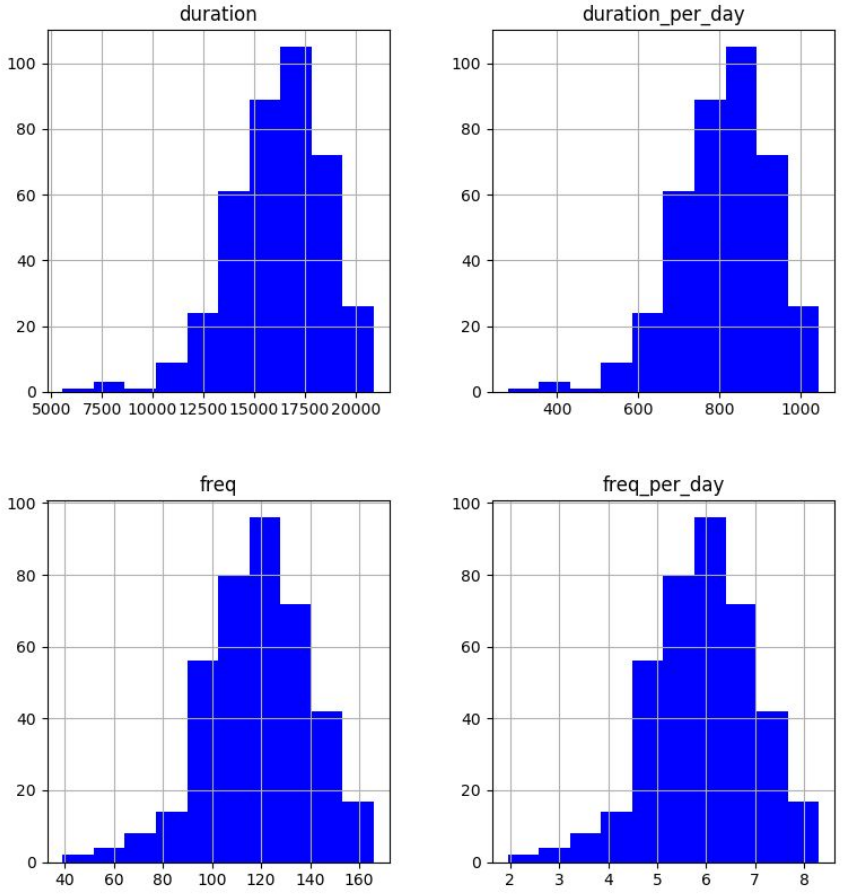
- **Bookings per day**
  - In average a **Car2go** has a cumulative booking time equal to **121 +- 28 min**
  - In average an **Enjoy** has a cumulative booking time equal to **236 +- 67 min**
- The results reported in the Andrea chicco's Thesis (*Analisi di Big Data relativi a servizi di car sharing: il caso di Car2Go a Torino*), based only on car2go:
  - Mean parking duration: 22h 31m

Previous Results I

Car2go:

- The (0,0) plot represents the PDF of parkings duration on all the period
- The (1,0) plot is the parkings frequence PDF on all the fleet on all period
- The (0,1) and (1,1) plots are obtained from the (0,0) and (1,0) dividing by the number of days with valid entries

- Median duration per day per car = 820 min
- Median booking freq. per day per car = 5.95

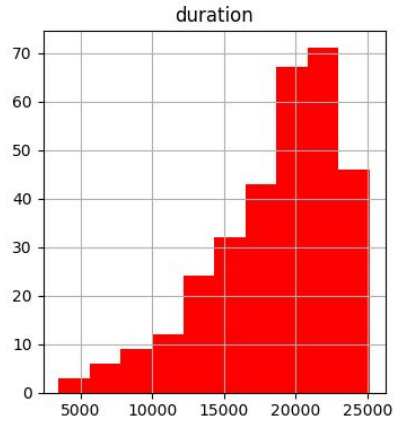


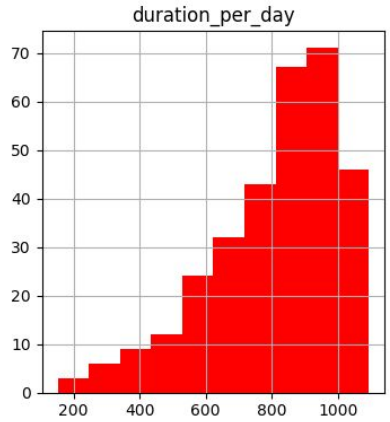
Previous Results II

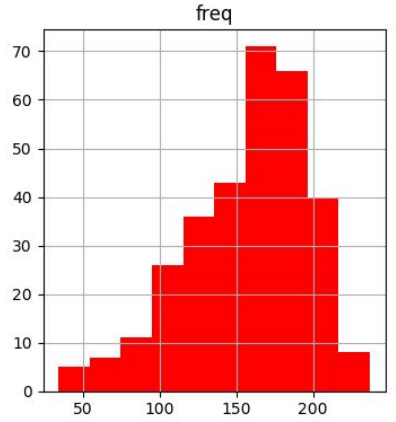
Enjoy:

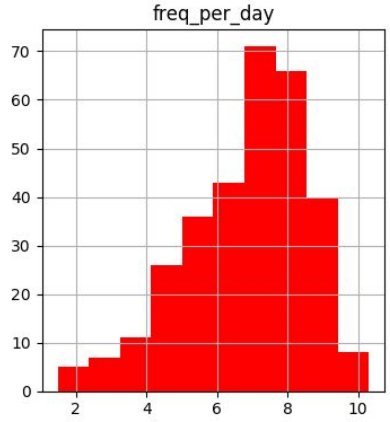
- The (0,0) plot represents the PDF of parkings duration on all period
- The (1,0) plot is the parking frequencies PDF on all the fleet on all period
- The (0,1) and (1,1) plots are obtained from the (0,0) and (1,0) dividing by the number of days with valid entries

- Median duration per day per car = 850 min
- Median parking freq. per day per car = 7.17









Previous results - corrections

- For all the following data sets are considered the data contained in the interval **[median - std dev; median+std_dev]** of a given metrics:
  - Valid days
  - Parking duration per car
  - Booking duration per car

Valid days - considerations

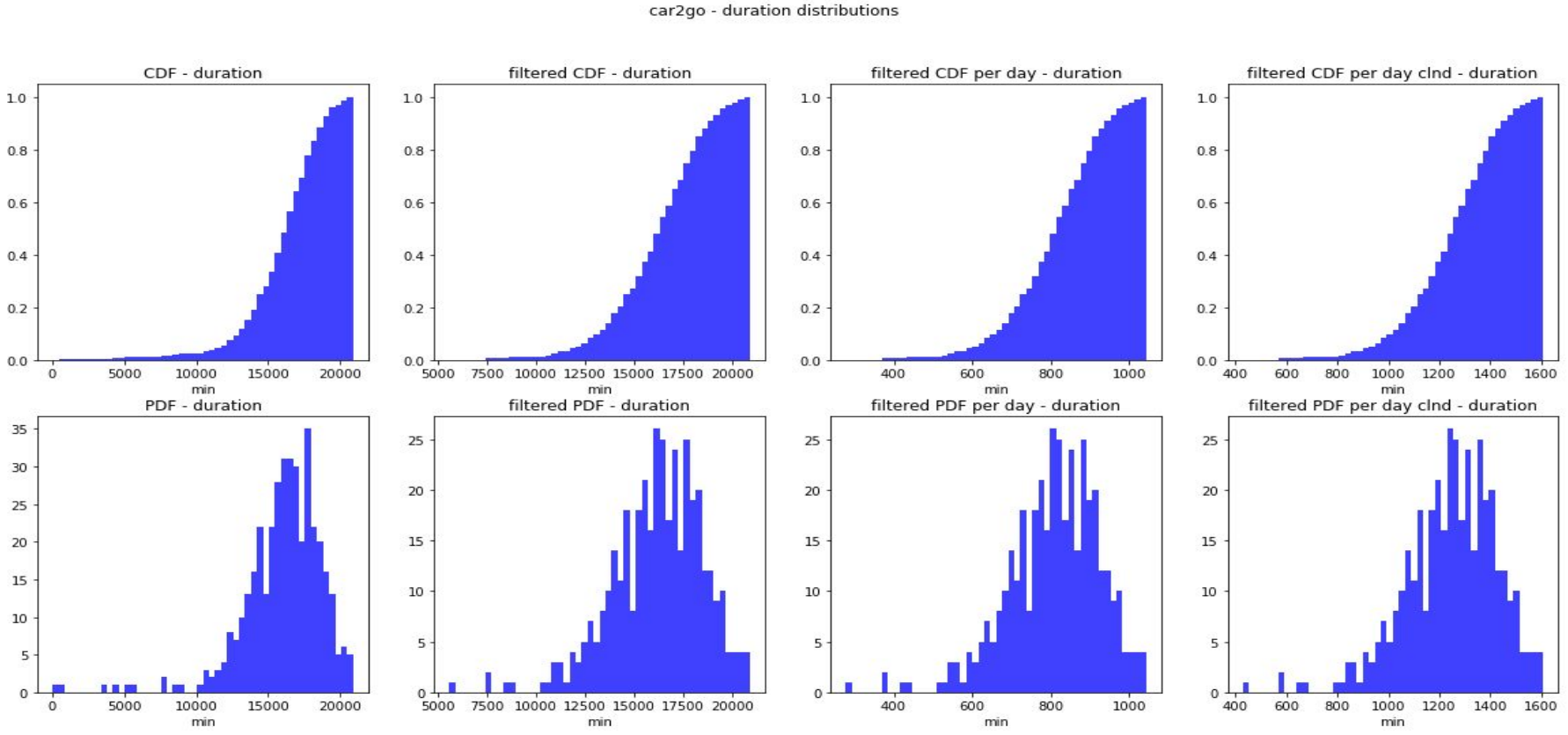
Car2go

- Valid days = 20
  - Days with entries
- Mean = 2123.4
- Median = 2012.5
- Std = 704.50
- Cleaned valid days = 13
  - Days with entries in interval [median-std; median +std]

Enjoy

- Valid days = 23
  - Days with entries
- Mean = 2029.91
- Median = 2048.0
- Std = 437.48
- Cleaned valid days = 18
  - Days with entries in interval [median-std; median +std]

Car2go - parkings duration

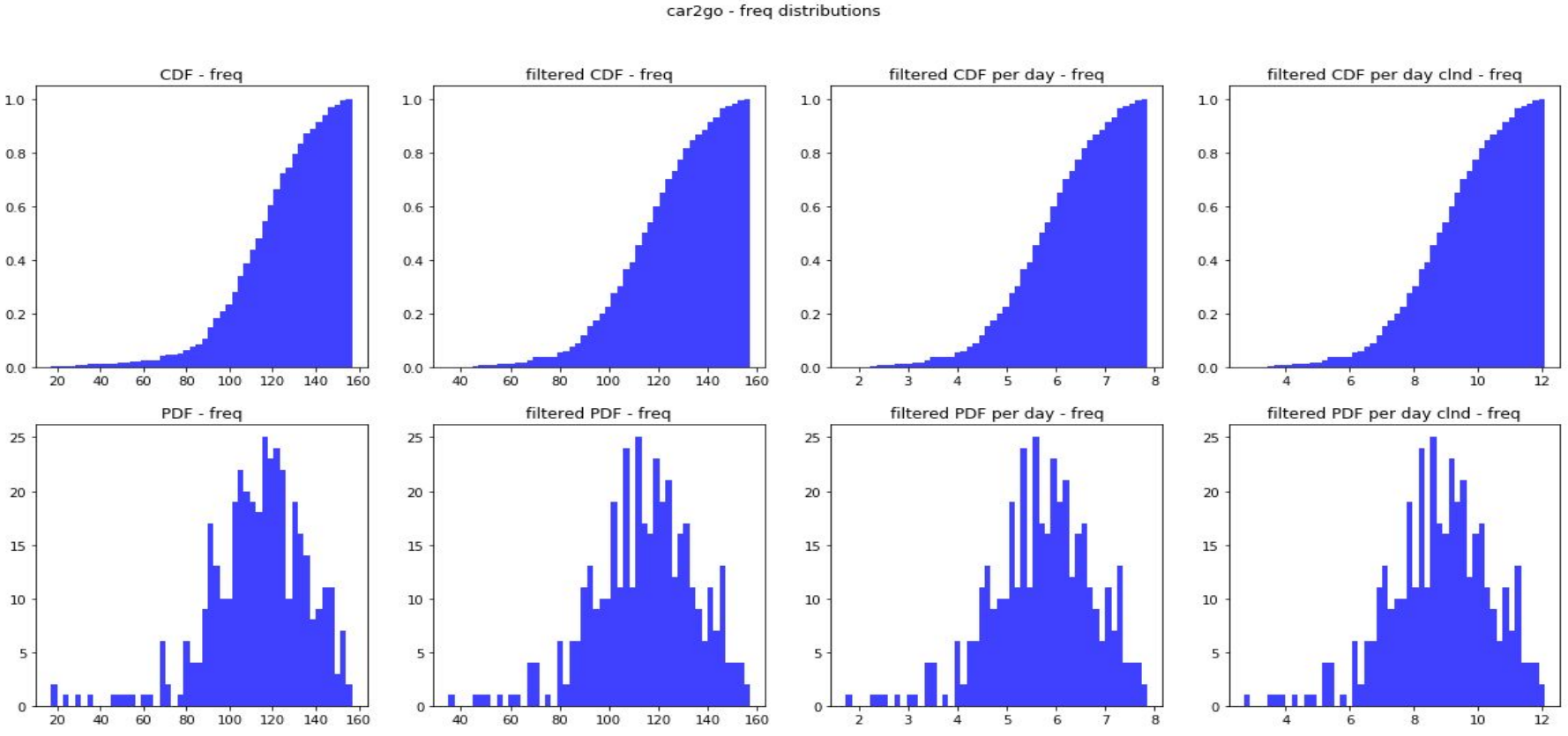


Car2go Parkings - duration

[min]	**Filtered duration***	**Per day****	**Per valid day*****
mean	16188.38	809.41	1245.26
median	16407.0	820.35	**1262.07**
std	2340.60	117.03	180.04

- * kept cars which have at least one parking per day
- ** obtained by dividing the distances by the number of days with entries
- *** obtained by dividing the distances by the number of days in the interval [median - std; median+std] of entries number

Car2go - parkings freq

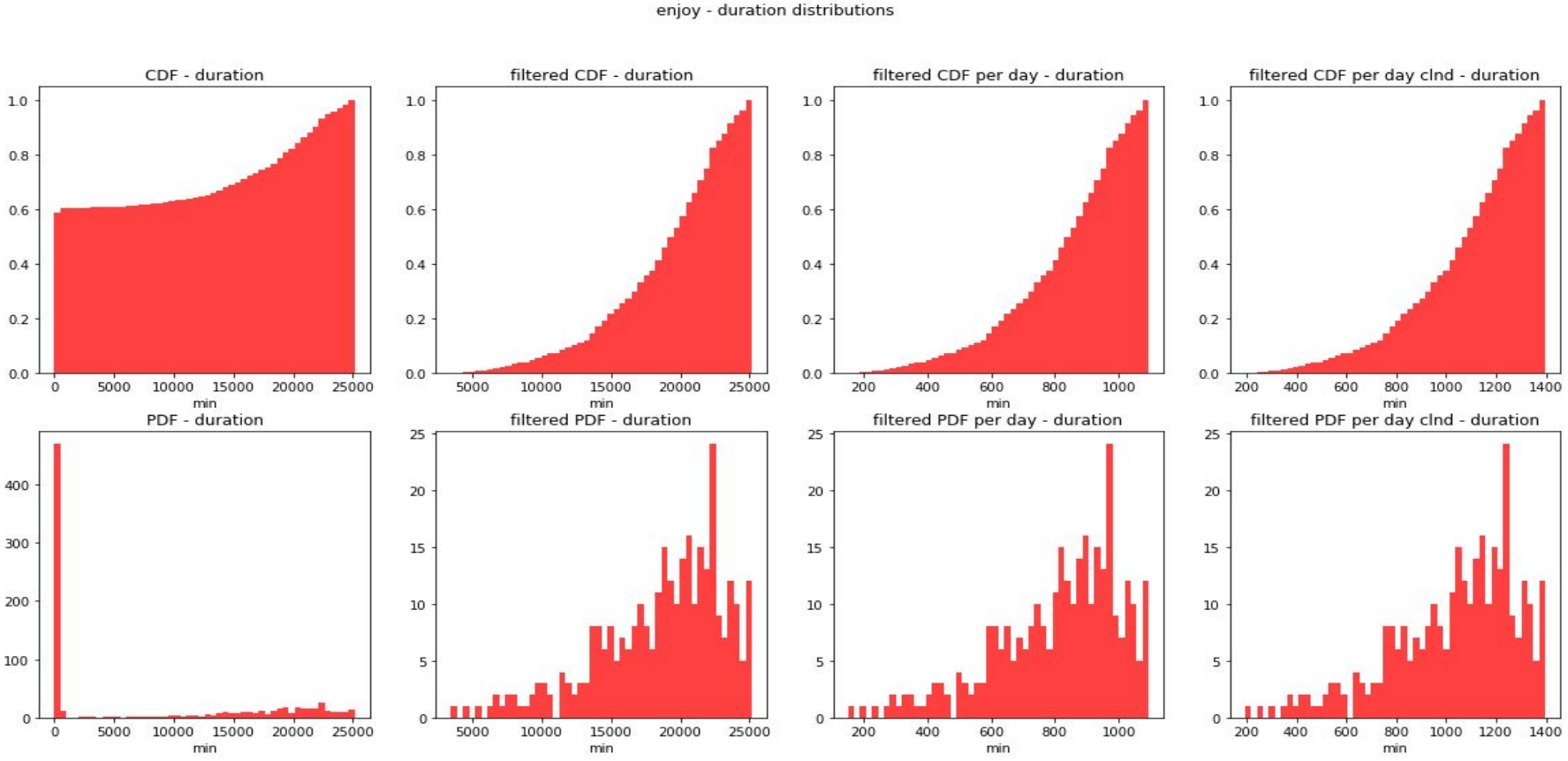


Car2go Parkings - freq

[-]	**Filtered freq***	**Per day****	**Per valid day*****
mean	114.38	5.71	8.71
median	115	5.75	**8.84**
std	20.57	1-02	1.58

- * kept cars which have at least one parking per day
- ** obtained by dividing the distances by the number of days with entries
- *** obtained by dividing the distances by the number of days in the interval [median - std; median+std] of entries number

Enjoy - parkings duration

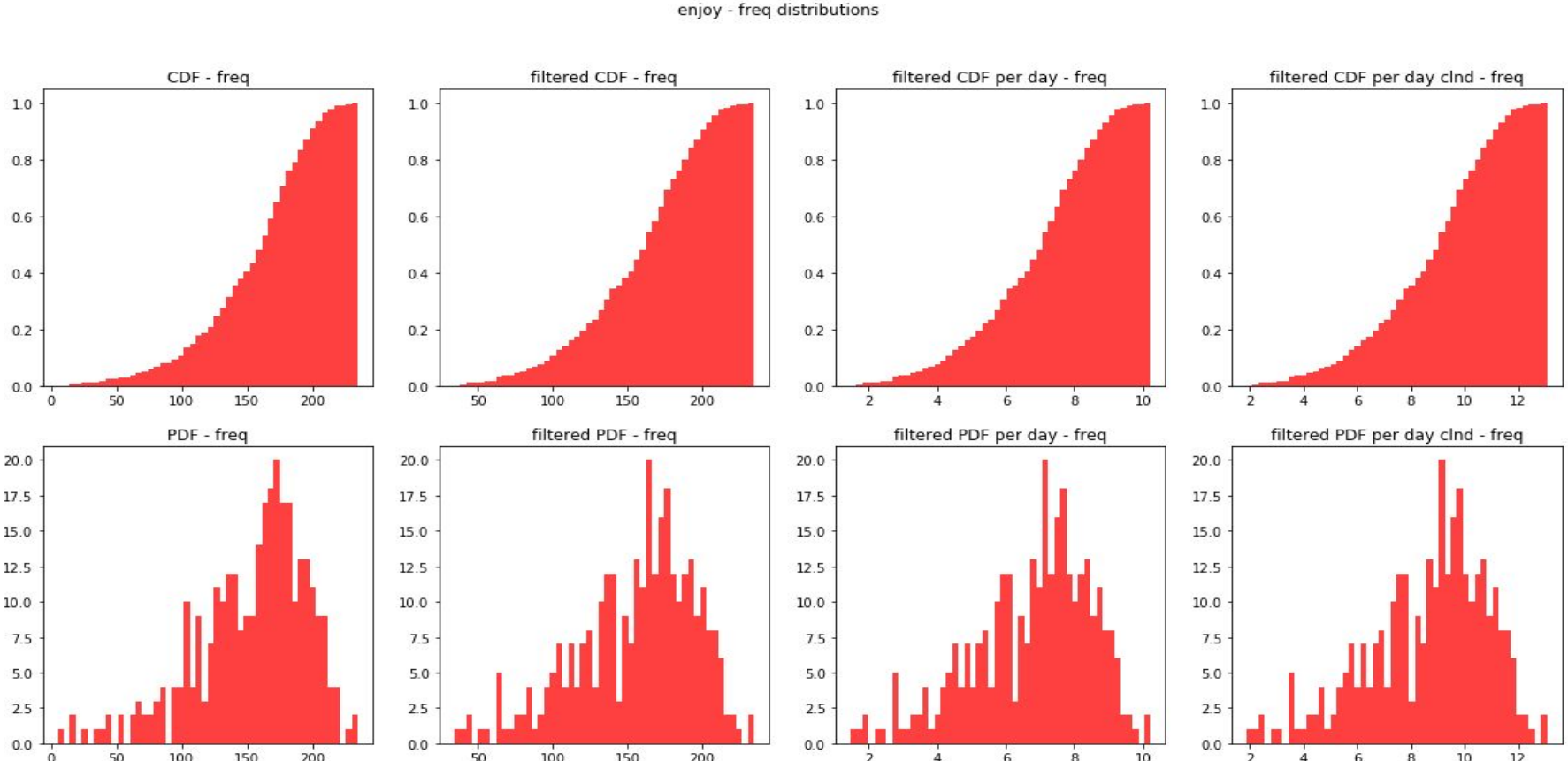


Enjoy Parkings - duration

[min]	**Filtered duration***	**Per day****	**Per valid day*****
mean	18637.96	810.34	1035.34
median	19555	850.21	**1086.38**
std	4493.77	195.38	249.65

- * kept cars which have at least one parking per day
- ** obtained by dividing the distances by the number of days with entries
- *** obtained by dividing the distances by the number of days in the interval [median - std; median+std] of entries number

Enjoy - parkings freq



Car2go Parkings - freq

[-]	**Filtered freq***	**Per day****	**Per valid day*****
mean	115.70	6.76	8.65
median	164	7.13	**9.11**
std	39.49	1.71	2.19

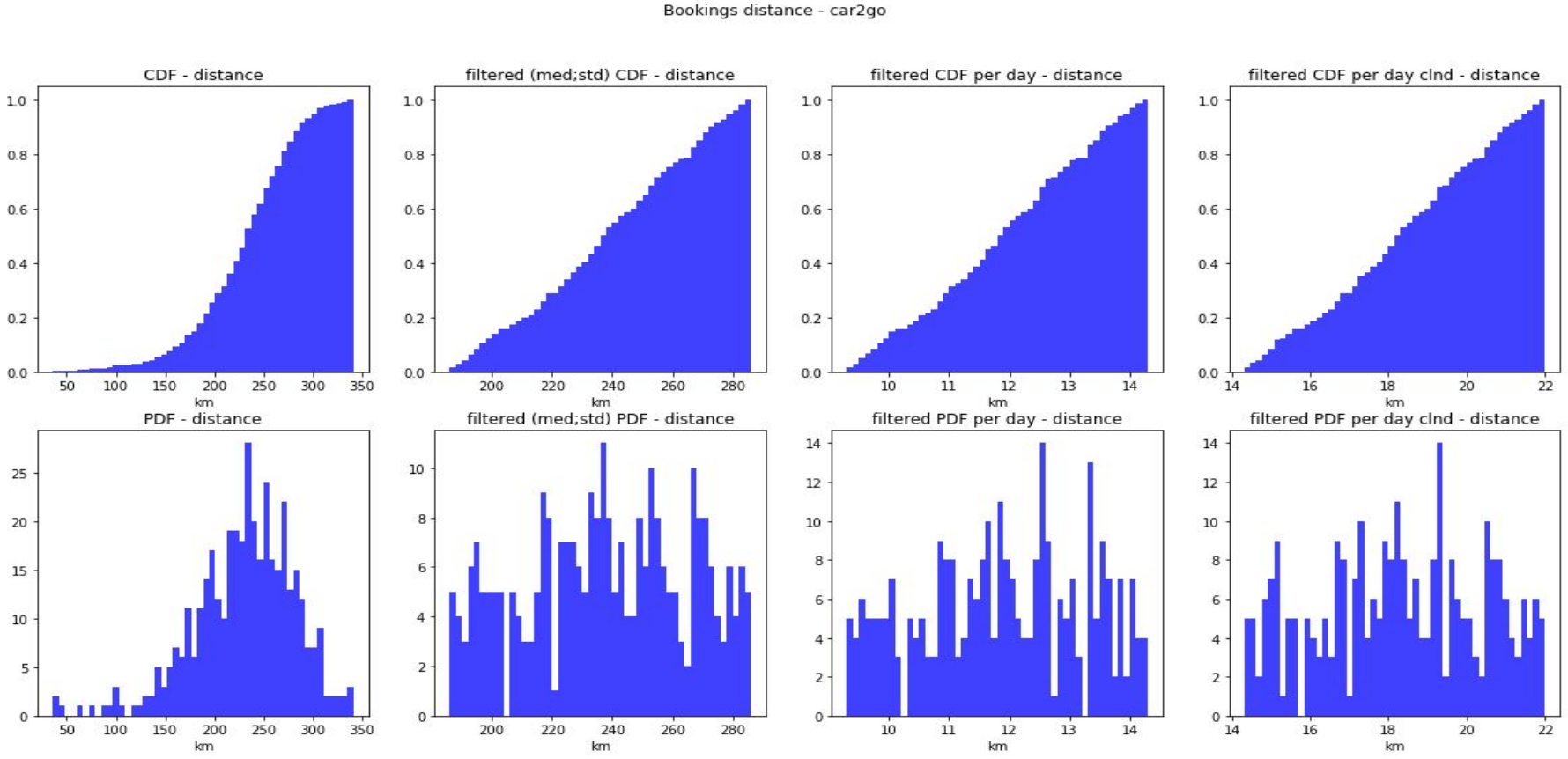
- * kept cars which have at least one parking per day
- ** obtained by dividing the distances by the number of days with entries
- *** obtained by dividing the distances by the number of days in the interval [median - std; median+std] of entries number

Considerations on parkings

Analyzing the period of one month:

- Parkings are underestimated
  - Weekdays and weekends days could have a difference of 2000 entries. Considering only one month, the number of days with few entries is comparable with the number of day with an high number of entries. In this way the average number of entries is bad estimated
  - By considering a bigger time period, this effect should be reduced
- Keeping the day inside in the interval [mean+-std]
  - Good estimation of parking time (according to the Chicco's study)
  - Few valid days: we keep 14 days on 31
- Analyzing a bigger time interval:
  - The difference between weekends and weekdays should be reduced
  - Cluster unreachable during this analysis

Car2go Bookings - distances

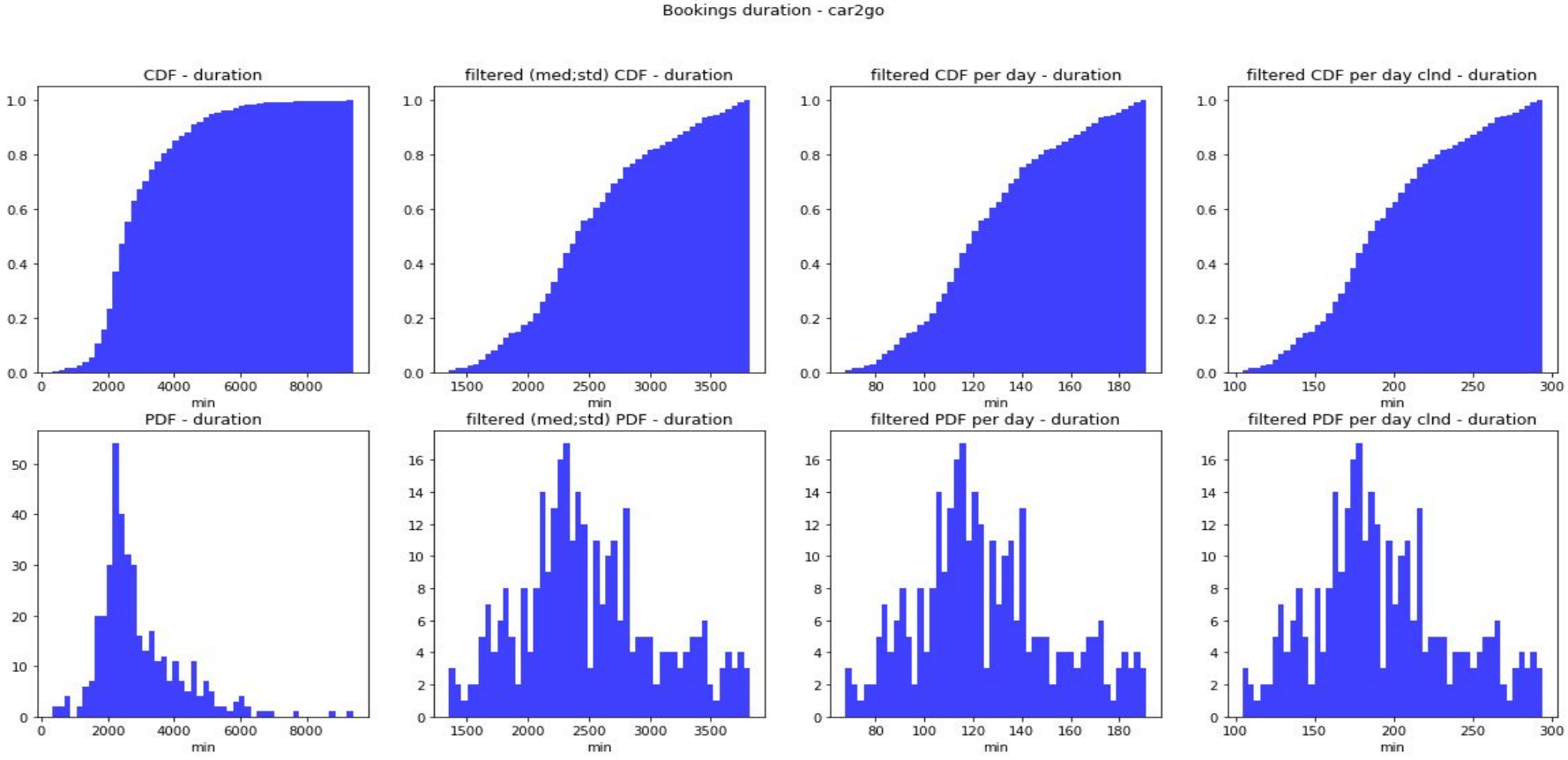


Car2go Bookings - distances

[km]	**unfiltered**	**filtered***	**Per day****	**Per valid day*****
mean	230.49	237.38	11.86	18.26
median	236.0	237.0	11.85	18.23
std	50.90	27.28	1.36	2.09

- * are kept samples in interval [median - std; median+std]
- ** obtained by dividing the distances by the number of days with entries
- *** obtained by dividing the distances by the number of days in the interval [median - std; median+std] of entries number

Car2go Bookings - durations

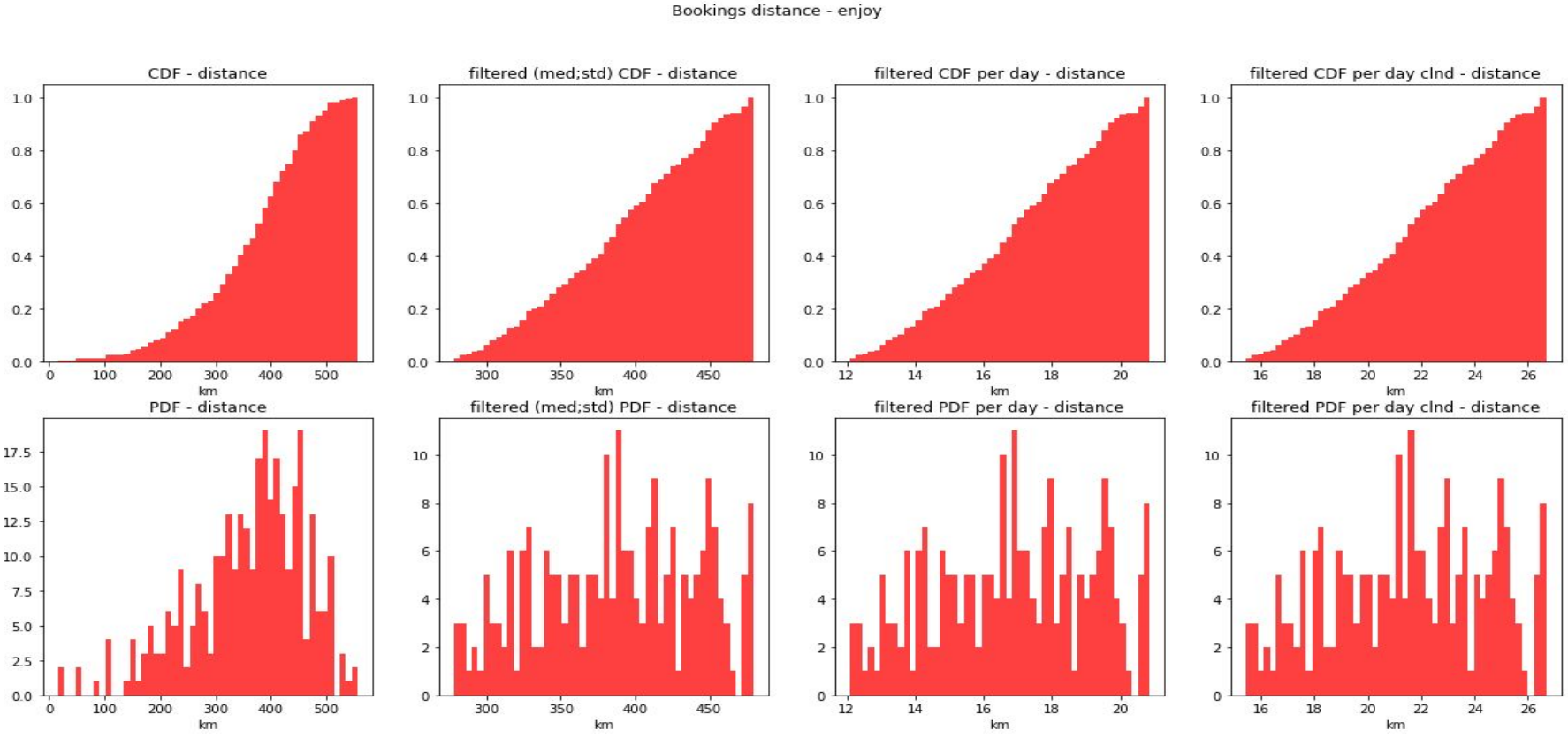


Car2go Bookings - distances

[min]	**unfiltered**	**filtered***	**Per day****	**Per valid day*****
mean	2927.56	2506.77	125.33	192.82
median	2585.0	2424.0	121.2	186.46
std	1242.35	565.55	28.27	43.50

- * are kept samples in interval [median - std; median+std]
- ** obtained by dividing the distances by the number of days with entries
- *** obtained by dividing the distances by the number of days in the interval [median - std; median+std] of entries number

Enjoy Bookings - distances

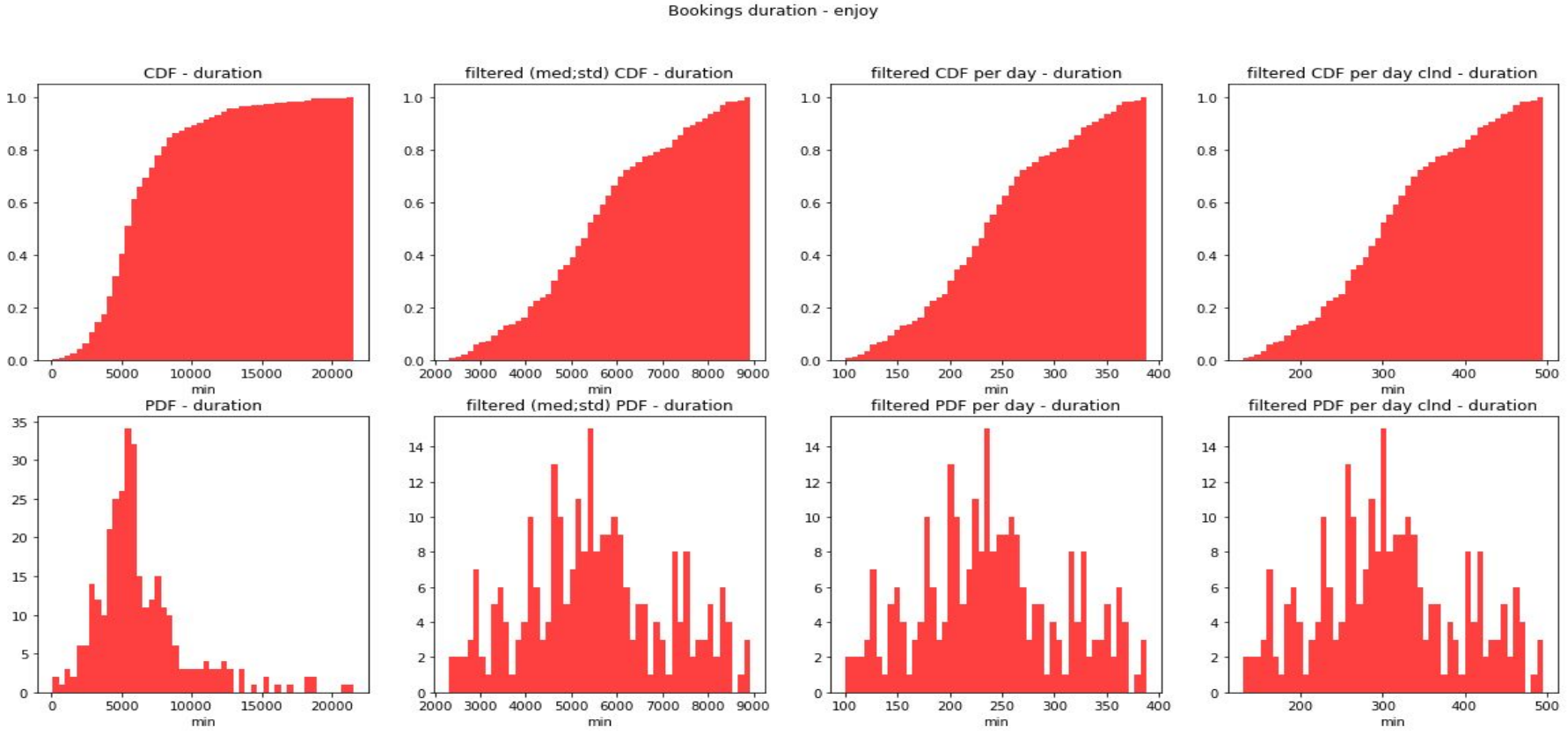


Enjoy Bookings - distances

[km]	**unfiltered**	**filtered***	**Per day****	**Per valid day*****
mean	359.99	387.52	16.84	21.52
median	379.0	390.0	16.95	21.66
std	102.89	53.12	2.30	2.95

- * are kept samples in interval [median - std; median+std]
- ** obtained by dividing the distances by the number of days with entries
- *** obtained by dividing the distances by the number of days in the interval [median - std; median+std] of entries number

Enjoy Bookings - duration



Enjoy Bookings - duration

[min]	**unfiltered**	**filtered***	**Per day****	**Per valid day*****
mean	6356.70	5527.97	240.34	307.10
median	5623.0	5428.5	236.02	301.58
std	3319.08	1547.21	67.27	85.95

- * are kept samples in interval [median - std; median+std]
- ** obtained by dividing the distances by the number of days with entries
- *** obtained by dividing the distances by the number of days in the interval [median - std; median+std] of entries number

Considerations on bookings

- Better parkings estimation time:
  - **24h - (mean bookings time)** to obtain a good estimation of mean parkings time per day
  - Coherent construction of parkings time
- Enjoy has a mean bookings time bigger than car2go.
  - 236 min per day vs 121 min per day
    - Due to a smaller fleet

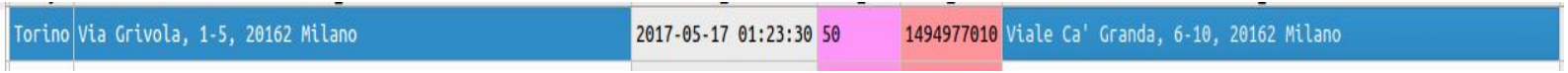
Analysis of car exchanging among cities

- Analysis period: from 17th May 2017 to 17th June 2017
- By querying all bookings, **half fleet is booked one times for few minutes**
  - Phenomenon visible only for Enjoy
- Plate identification of cars with strange behaviour:
  - All the cars which have a cumulative distance lower than 30 km (in Torino) *
- Query the cars which respect the * condition per plate
- Saved in format
  - **[Key: (plate, city); Value: number of bookings for those car in those city (unplottable)]**
  - **Sum all the bookings, for the cars in** *, **for each city** (in the next slide)
    - Is possible to notice that 547 cars are re-injeceted in Torino or are “jolly cars”
- Any of those cars is in the Active Bookings collections

Analysis of car exchanging among cities

Index	ookings_per_ca	log
Catania	309	2.49
Firenze	815	2.91
Milano	49481	4.69
Roma	1506	3.18
Torino	547	2.74

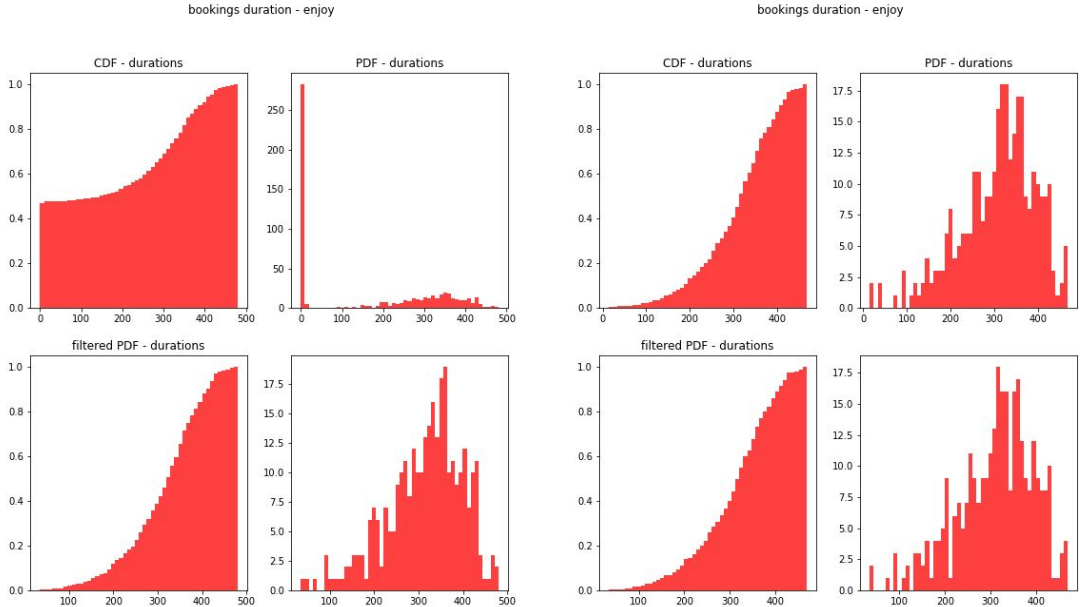
Improvements and corrections - Crawler error

- The previous data come from a scraping error
- 

- In the Database proxy is added a constraint on the initial and final address: a regex expression decide which entries to take and which not
  - **(nn… city)** where:
    - **nn** first two digit of CAP of the city parameter
    - **'…'** means each number from 0 to 9
    - **city** the city passed as parameter
- Not scalable on other cities, but parameterizable
  - It is possible check also the coordinates of booking
- Analysis done on Enjoy, Car2go not yet analyzed

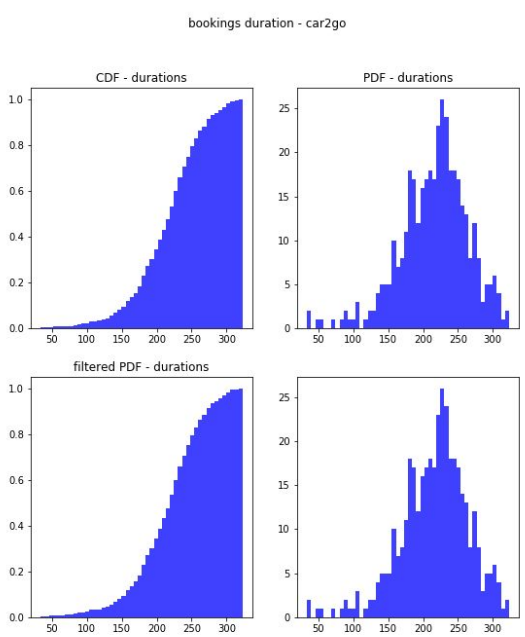
Improvements and corrections

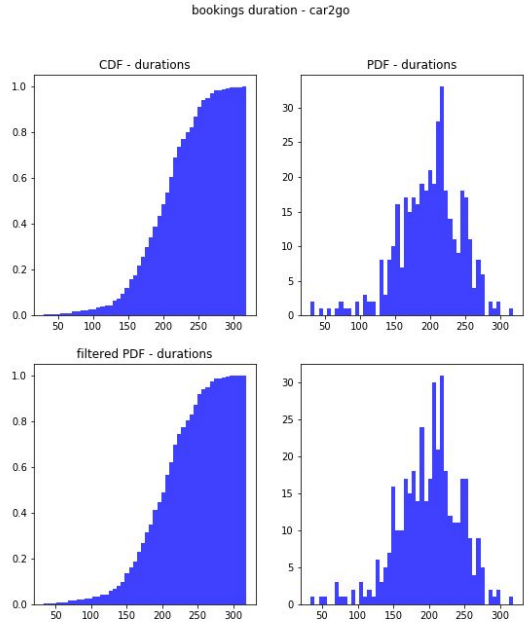
- Keeping the entries which have initial and final address in the same city of the city field, the outlier are discarded
  - on the left: biased data
  - on the right: unbiased data
  - The data are first and before filtering
  - **wrong title: is distance** not duration



Improvements and corrections

- The mismatch phenomenon between address and city has less influence on Car2go
  - **wrong title: is distance** not duration



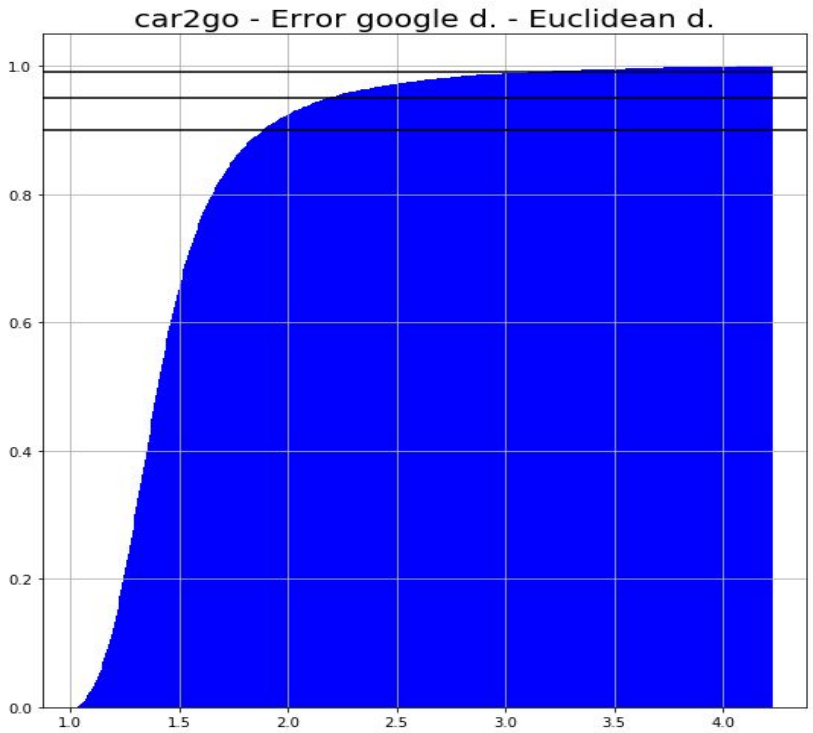


Google distances (g.d.) vs Euclidean distance (e.d.)

- Took all the cars with at least 20m of e.d. and a g.d. different from -1
  - Enjoy lost 36698 on a total of 50165 (73%): the scraper could have some problems
    - Cookies issues
  - Cargo lost 11923 on a total of 46437 (25%)
- The g.d. is divided by the e.d. for all entries
- Then is computed the CDF in order to obtain a corrective factor to apply to e.d.

Car2go - corrective factor

0.9	0.95	0.99
1.88907784116	2.19166631065	3.14372600959

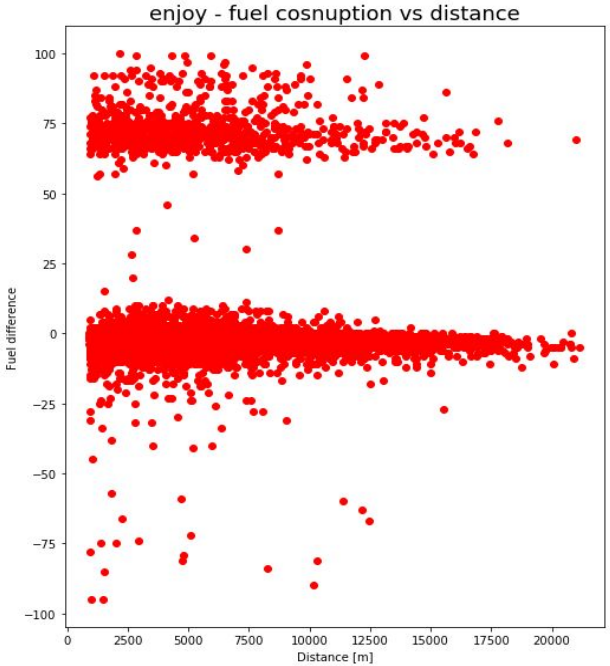


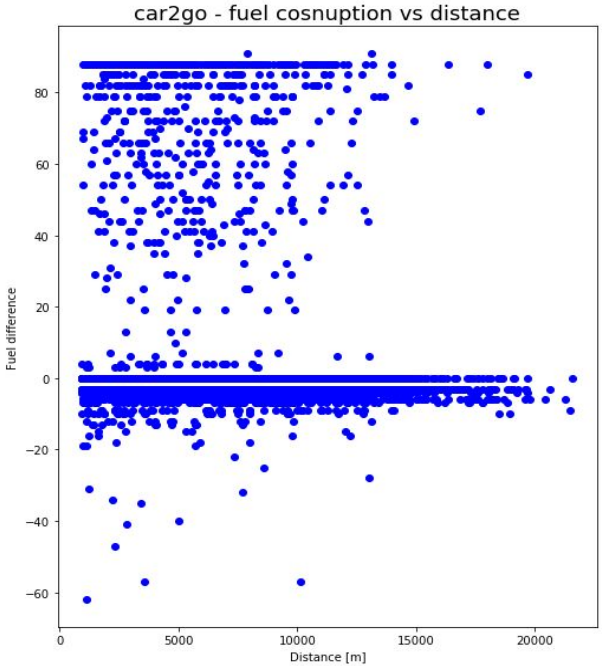
Enjoy - corrective factor

0.9	0.95	0.99
1.82844597649	2.15828664066	3.59500679385

Fuel consumption analysis

- clean dataset
  - No entries with Milan city but different address
- Y: fuel difference
  - Final fuel - initial fuel
  - Refill > 0
  - Consumption < 0
- X: e.d. * corrective factor
- Distance > 500m





- Is possible to notice that Enjoy gives incentives to users to refill the car during their run

Parking time per zone

- Took all parkings, and are **counted** how many parkings per zone
- Took all parkings and are **summed** all the **duration** per each zone
- Then the sum of the zonal parking duration are multiplied per the number of parking in the same zone
  - Called ***factor***
- Factor divided per:
  - 60: obtaining the hour from the total amount of minutes
  - Number of parkings in the area, to obtain the **average parking time for the area**

Average parking time for each zone