Browse Till You Die: Hierarchical Scalable Model of Cookie Deletion

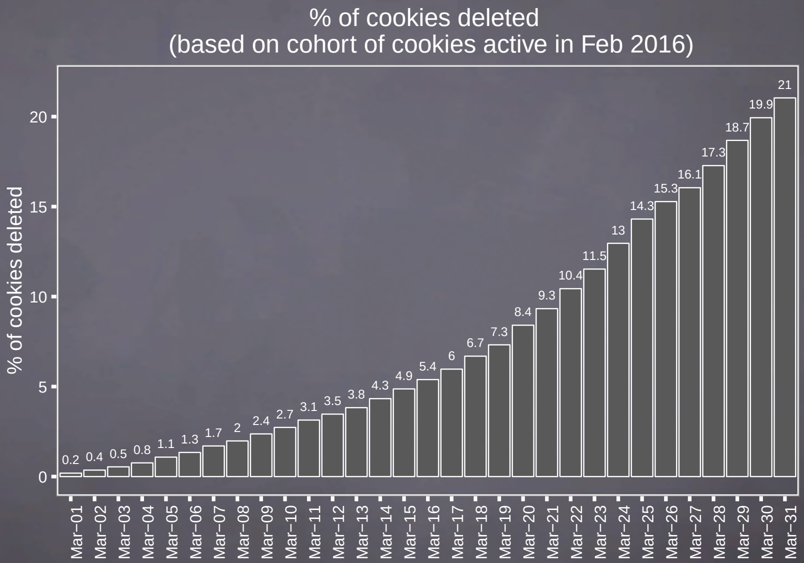
Jakub Glinka, GfK Data Lab
jakub.glinka@gfk.com

eRum 2016, 13 October

Agenda

- Problem description
- Model Based Machine Learning
- Browse Till You Die
    - Assumptions
    - Estimation
    - Prediction
    - Accuracy
- Summary

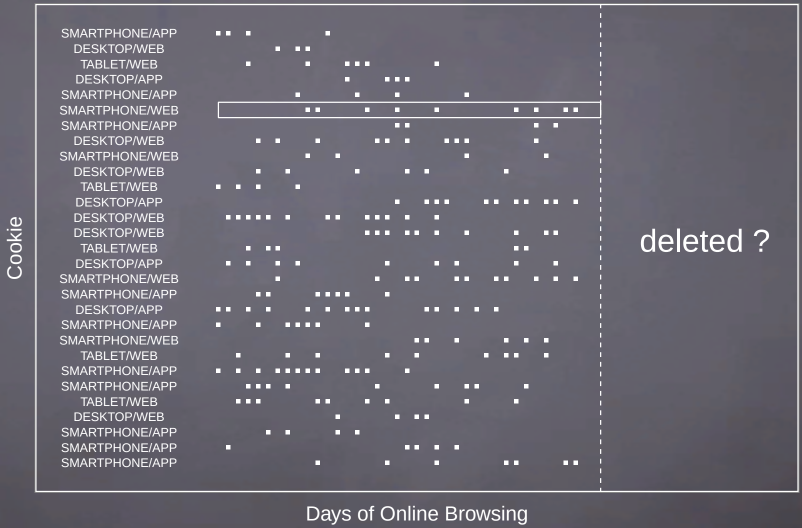
Problem description

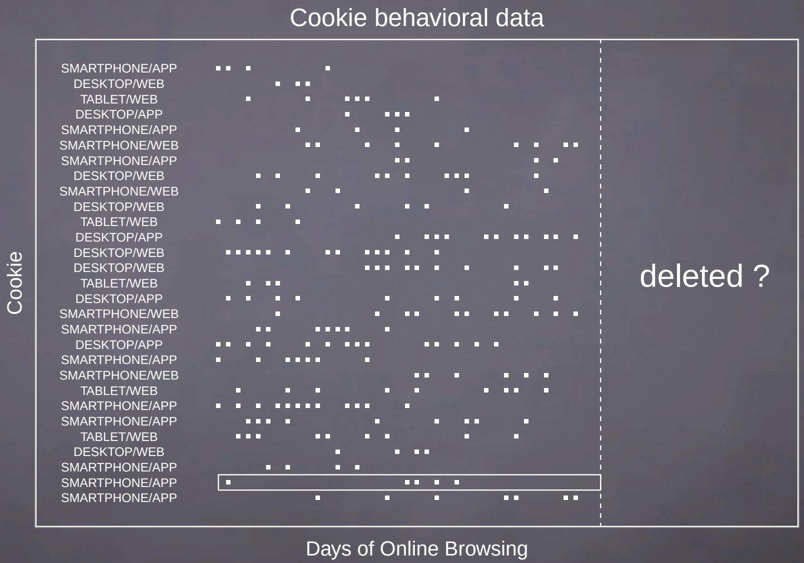

% of cookies deleted
(based on cohort of cookies active in Feb 2016)
% of cookies deleted
0
5
10
15
20
0.2
0.4
0.5
0.8
1.1
1.3
1.7
2
2.4
2.7
3.1
3.5
3.8
4.3
4.9
5.4
6
6.7
7.3
8.4
9.3
10.4
11.5
13
14.3
15.3
16.1
17.3
18.7
19.9
21
Mar−01
Mar−02
Mar−03
Mar−04
Mar−05
Mar−06
Mar−07
Mar−08
Mar−09
Mar−10
Mar−11
Mar−12
Mar−13
Mar−14
Mar−15
Mar−16
Mar−17
Mar−18
Mar−19
Mar−20
Mar−21
Mar−22
Mar−23
Mar−24
Mar−25
Mar−26
Mar−27
Mar−28
Mar−29
Mar−30
Mar−31

Cookie behavioral data

Cookie behavioral data

Cookie behavioral data




Cookie behavioral data
SMARTPHONE/APP
DESKTOP/WEB
TABLET/WEB
DESKTOP/APP
SMARTPHONE/APP
SMARTPHONE/WEB
SMARTPHONE/APP
DESKTOP/WEB
SMARTPHONE/WEB
DESKTOP/WEB
TABLET/WEB
DESKTOP/APP
DESKTOP/WEB
DESKTOP/WEB
TABLET/WEB
DESKTOP/APP
SMARTPHONE/WEB
SMARTPHONE/APP
DESKTOP/APP
SMARTPHONE/APP
SMARTPHONE/WEB
TABLET/WEB
SMARTPHONE/APP
SMARTPHONE/APP
TABLET/WEB
DESKTOP/WEB
SMARTPHONE/APP
SMARTPHONE/APP
SMARTPHONE/APP
deleted ?
Cookie
Days of Online Browsing

Standard Machine Learning solution drawbacks

- due to unobserved deletion time we need to wait certain period of time to be able to asses whether cookie was deleted or not
- for some cookies amount of information is very sparse
- till the moment we will be applying our model it might be already outdated

Model Based Machine Learning

Goals of model-based machine learning [1]:

- ability to create a very broad range of models
- segregation between the model and the inference: if changes are made to the model, the corresponding inference software is created automatically
- transparency of functionality: the model is described by compact code within a generic modeling language
- pedagogy: newcomers to the field of machine learning have only to learn single modeling environment in order to be access a wide range of modelling solutions

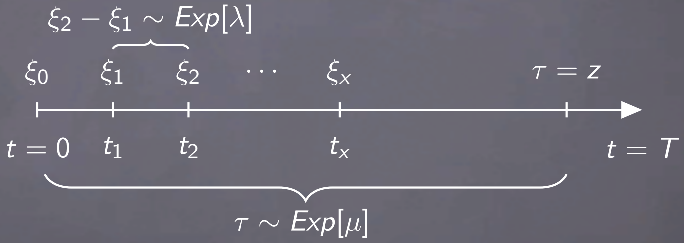
[1] Bishop CM. 2013 Model-based machine learning. http://dx.doi.org/10.1098/rsta.2012.0222

Model-based ML recipe

- Step1:
  express problem in the form of probabilistic graphical model
- Step2:
  specify data and parameters likelihood
- Step3:
  implement model in probabilistic programming language

Browse Till You Die

Data generation model - individual cookie level[2]



[2]Counting Your Customers David C. Schmittlein, Donald G. Morrison and Richard Colombo, Management Science (1987)

Data generation model - cookie likelihood

$$\mathbb{P}[(\xi_1 = t_1, ..., \xi_x = t_x, \tau = z, T)] =$$

$$\mathbb{P}[\tau = z] \qquad \xi_x \sim Exp[\mu]$$

$$\times$$

$$\mathbb{P}[\xi_x = t_x] \qquad \xi_x \sim \Gamma[x, \lambda]$$

$$\times$$

$$\mathbb{P}[(\xi_1 = t_1, ..., \xi_{x-1} = t_{x-1})|\xi_x = t_x] \qquad C(x) = \frac{\Gamma(x)}{t^{x-1}}$$

$$\times$$

$$\mathbb{P}[\xi_{x+1} > z|\xi_x = t_x] \qquad \exp(-\lambda * (min\{z, T\} - t_x))$$

Data generation model - parameters likelihood

We assume hierarchical likelihood over cookie individual parameters:

- hyper-parameters prior:
  - $r \sim \Gamma(1e-3, 1e-3)$
  - $\alpha_j \sim \mathrm{Normal}(0, 100)$
  - $s \sim \Gamma(1e-3, 1e-3)$
  - $\beta \sim \Gamma(1e-3, 1e-3)$

- parameters prior:
  - $(\lambda_i | r, \alpha) \sim \Gamma(r, \exp(-d^\top \alpha))$
  - $(\mu_i | s, \beta) \sim \Gamma(s, \beta)$

Stan implementation

```
r ~ gamma(1e-3,1e-3);                  // hyper-parameters
s ~ gamma(1e-3,1e-3);                  // likelihood
alpha ~ normal(0.0,100.0);
beta ~ gamma(1e-3,1e-3);

lambda ~ gamma(r,exp(-alpha*D)); // parameters
mu ~ gamma(s,beta);                    // likelihood

t + tau ~ exponential(mu);             // complete data
for (i in 1:N) {                       // likelihood
  if (x[i] > 0) {
    t[i] ~ gamma(x[i],lambda[i]);
  }
  target += -lambda[i]*fmin(tau[i],T[i]-t[i]);
}
```

Interesting theoretical model properties

- possible to fit on the current batch of data
- easy updating of the parameters (stability over time)
- likelihood depends only on (d, x, t, T) - scalability

Model Estimation

Consensus Monte Carlo[3]

1. Divide y into shards $y_1, ..., y_S$
2. Run S separate MCMC algorithms to sample $\theta_s \sim p(\theta|y_s)$ for each shard using the fractionated prior $p(\theta)^{1/S}$
3. Combine the draws across shards using weighted averages:

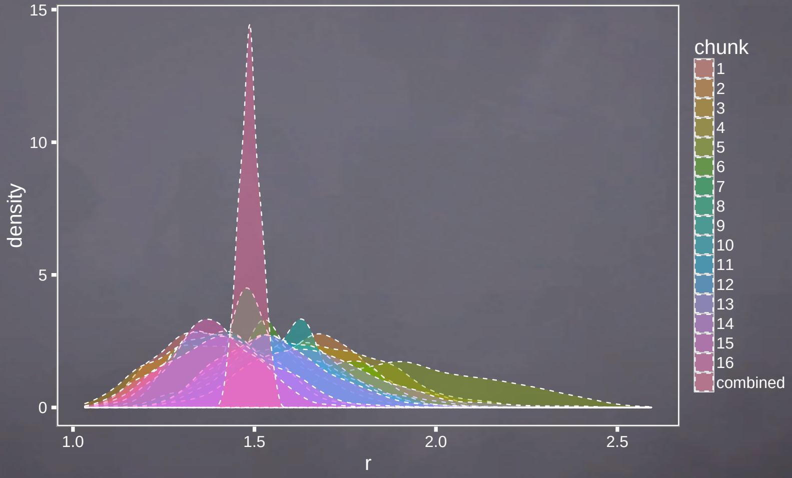
$$\theta = (\sum_s W_s)^{-1}(\sum_s W_s\theta_s)$$

with common choice being:

$$W_s = \Sigma_s^{-1},\ \Sigma_s = Var(\theta|y_s)$$

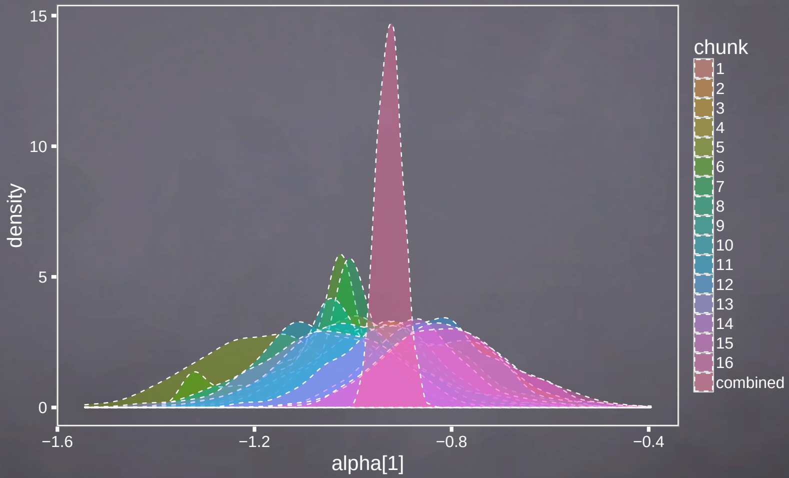
[3] Bayes and Big Data: The Consensus Monte Carlo Algorithm Steven L. Scott et.al. 2013

Consensus Monte Carlo results (N=10k, S=16)



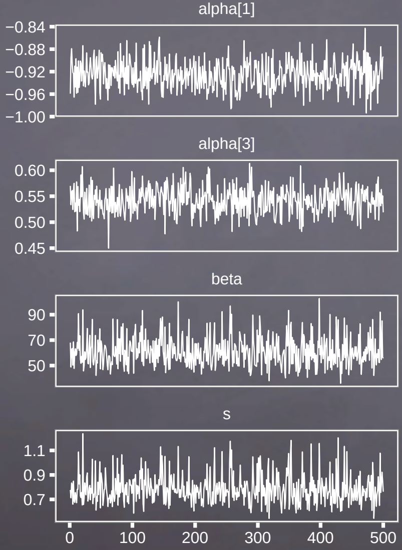
posterior obtained using parallelMCMCcombine package

Consensus Monte Carlo results (N=10k, S=16)



posterior obtained using parallelMCMCcombine package

Consensus Monte Carlo results: Trace plot



Posterior distribution of hyperparameters

Device	Context	r	α	$r\exp(\alpha)$	avg$[\frac{x}{T}]$
DESKTOP	WEB	1.49	-1.67	0.28	0.24
DESKTOP	APP	1.49	-0.93	0.59	0.5
SMARTPHONE	WEB	1.49	-2.04	0.19	0.13
SMARTPHONE	APP	1.49	-1.29	0.41	0.4
TABLET	WEB	1.49	-1.13	0.48	0.44

Device	Context	s	β	$\frac{s}{\beta}$	avg$[\frac{1}{T}]$
ALL	ALL	0.79	61.26	0.01	0.03

... so no prior aging of the cookies.

Prediction

Posterior distribution of individual parameters

Let's assume that we see new cookie with parameters d, x, t, T. We need to sample individual level posterior for λ, μ and τ.

Luckily posterior distributions have closed form in this particular model...

Still we want fast prediction function for our model hence: Rcpp!

Rcpp algorithm outline (single iteration)

Given $r^{(m)},\ \alpha^{(m)},\ s^{(m)},\ \beta^{(m)}$ from consensus posterior and $\lambda^{(m-1)},\ \mu^{(m-1)}$ from previous iteration:

$S_i \sim \mathbb{P}[\tau > T | x, t, T, \lambda^{(m-1)}, \mu^{(m-1)}]$

if $S_i = 0$ **then**

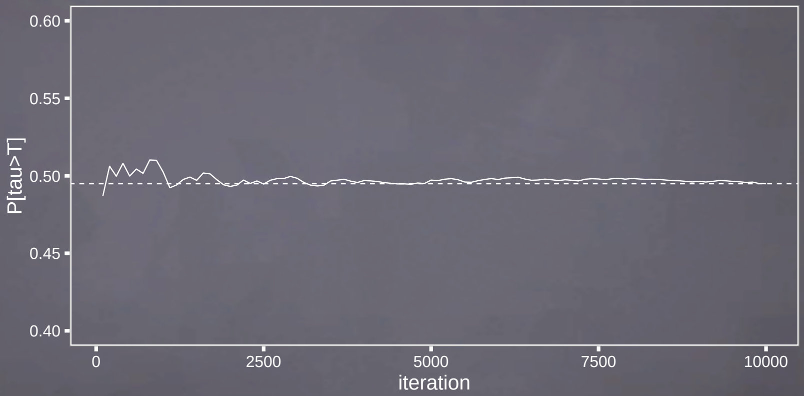
$\tau^{(m)} \sim Exp(\lambda^{(m-1)} + \mu^{(m-1)})|_{t,T}$
$\mu^{(m)} \sim \Gamma(s^{(m-1)} + 1, \beta^{(m-1)} + \tau^{(m)})$
$\lambda^{(m)} \sim \Gamma(r^{(m-1)} + x, \exp(-d^\top \alpha^{(m-1)}) + \tau^{(m)})$

else

$\tau^{(m)} \sim Exp(\mu^{(m-1)})|_{T,\infty}$
$\mu^{(m)} \sim \Gamma(s^{(m-1)} + 1, \beta^{(m-1)} + \tau^{(m)})$
$\lambda^{(m)} \sim \Gamma(r^{(m-1)} + x, \exp(-d^\top \alpha^{(m-1)}) + T)$

end if

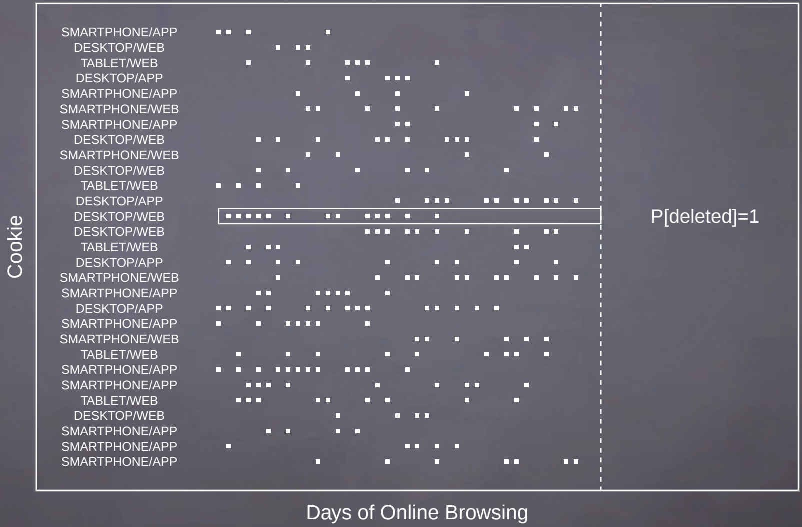
Monte Carlo Integration convergence:

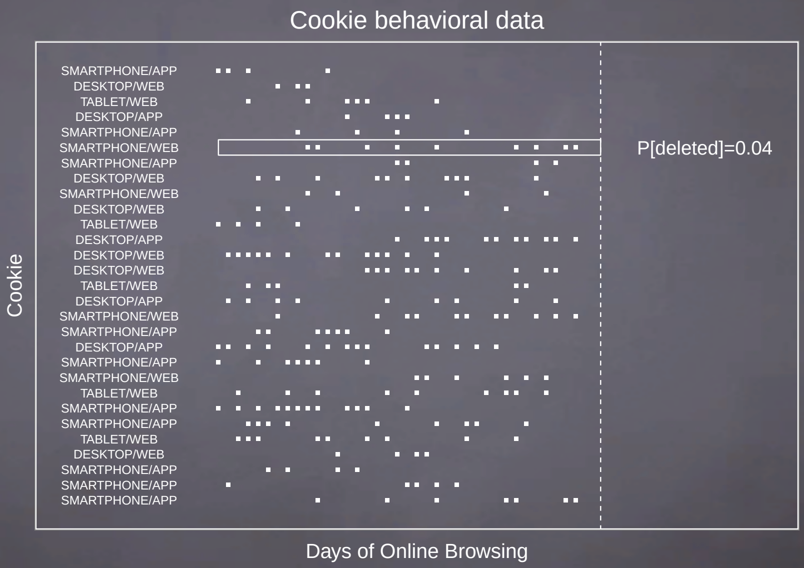


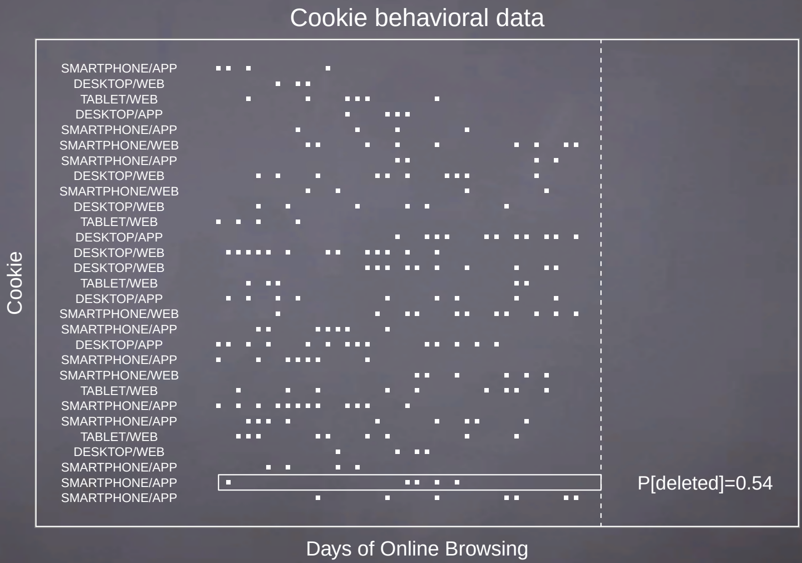

```
## Time of prediction for 1000 cookies:

##    user  system elapsed
##    0.56    0.00    0.56
```

Finaly we can score our database

Cookie behavioral data

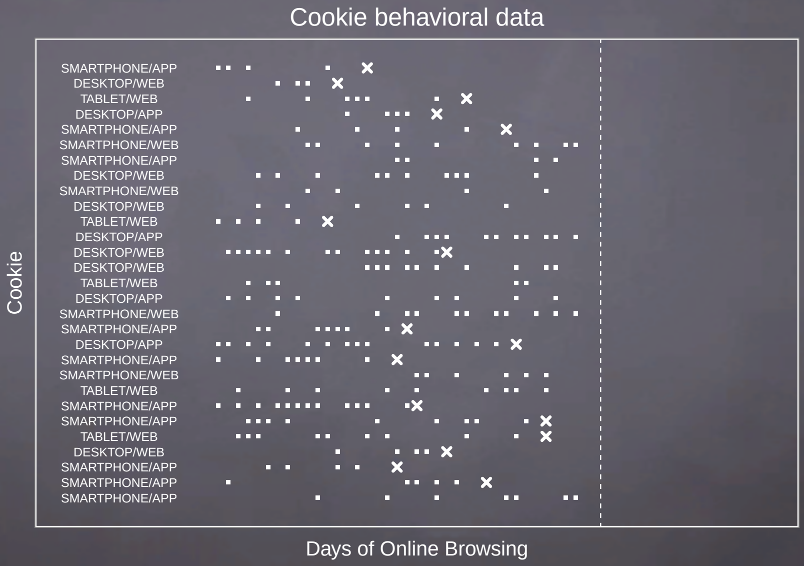



Cookie behavioral data
Cookie
SMARTPHONE/APP
DESKTOP/WEB
TABLET/WEB
DESKTOP/APP
SMARTPHONE/APP
SMARTPHONE/WEB
SMARTPHONE/APP
DESKTOP/WEB
SMARTPHONE/WEB
DESKTOP/WEB
TABLET/WEB
DESKTOP/APP
DESKTOP/WEB
DESKTOP/WEB
TABLET/WEB
DESKTOP/APP
SMARTPHONE/WEB
SMARTPHONE/APP
DESKTOP/APP
SMARTPHONE/APP
SMARTPHONE/WEB
TABLET/WEB
SMARTPHONE/APP
SMARTPHONE/APP
TABLET/WEB
DESKTOP/WEB
SMARTPHONE/APP
SMARTPHONE/APP
SMARTPHONE/APP
P[deleted]=0.04
Days of Online Browsing

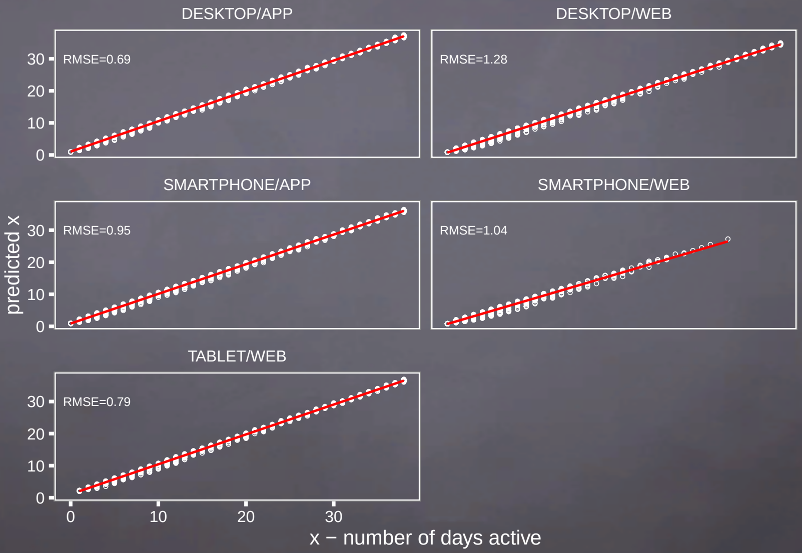

Cookie behavioral data
Cookie
SMARTPHONE/APP
DESKTOP/WEB
TABLET/WEB
DESKTOP/APP
SMARTPHONE/APP
SMARTPHONE/WEB
SMARTPHONE/APP
DESKTOP/WEB
SMARTPHONE/WEB
DESKTOP/WEB
TABLET/WEB
DESKTOP/APP
DESKTOP/WEB
DESKTOP/WEB
TABLET/WEB
DESKTOP/APP
SMARTPHONE/WEB
SMARTPHONE/APP
DESKTOP/APP
SMARTPHONE/APP
SMARTPHONE/WEB
TABLET/WEB
SMARTPHONE/APP
SMARTPHONE/APP
TABLET/WEB
DESKTOP/WEB
SMARTPHONE/APP
SMARTPHONE/APP
SMARTPHONE/APP
P[deleted]=0.54
Days of Online Browsing

But we can much more:

Prediction of exact moment of deletion:



Predicted number of days active



Accuracy

Model Accuracy (validated using additional month of data)

```
## Area under the curve: 0.9084

## $diag
## [1] 0.8551
##
## $kappa
## [1] 0.6990987

##
##          alive   deleted
##   0 0.8369772 0.1630228
##   1 0.1138829 0.8861171
```

Thank You!