3/16

Recommendation

Neighborhood Method:

1) (user, user) measure
2) (item, item) measure

Pros: No trainning

Challenges: PPl rate differently for different reasons
shift over time
upfront price for user might be high

Feature Extraction - Content-Based

reg. parameter

Similar to SVD where we decompose.

$$p^{(j)} = \underset{p}{\operatorname{argmin}} \frac{1}{||m^{(j)}||} \sum_{i \in m^{(j)}} (p^T q^{(i)} - r_{ij})^2 + \underbrace{\lambda ||p||^2}_{\text{Regularization}}$$

We don't use SVD bc we have unknown values in our matrix

Can we provide prediction w/o train feature: learn P & Q at the same time

Regularization for both term

Library we might use in midterm:

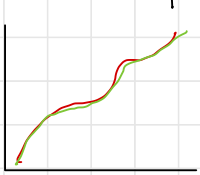
"scikit-surprise" filtering → result might be very large & very sparse

> Linear Regression Challenge:
> predict when alarm will go off using Linear Regression

Logistic Regression

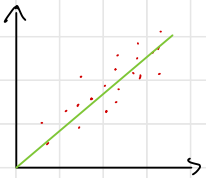
① define a cost function to minimize the distance between h(xi), yi.

when we have plot like this:



It's overfitting (i.e. memorize the model)

Should be:



Best fit

linear, but has randomness around it

② In terms of probability: maximize the probability observe the data $L(h) = P(\mathcal{D}|h)$

These 2 above are the method we use i.e. maximize probability, or minimize loss function

Assumptions: $\vec{y} = h_\beta(X) + \vec{\varepsilon}$

① linear function

② noise

Least-Square (distance perspective)

$\beta_{LS} = (X^T X)^{-1} X^T y$

Maximum Likelihood (probability approach)

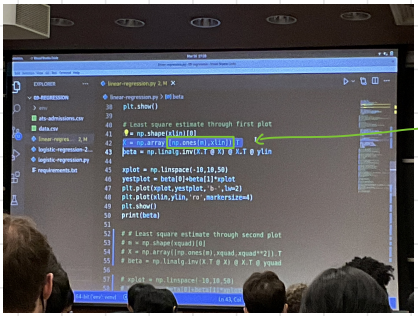
$\beta_{MLE} = \beta_{LS} = (X^T X)^{-1} X^T y$

give the same result

An unbiased Estimator

$E[\beta_{LS}] = \beta$

Code:



we want to get the constant

so not only using x_1 but the 1

#plot quadratic

quad = [1, a3, 5] ← 3 weights