# CS578 – Interactive and Transparent Machine Learning

# Topic: Regression

**Mustafa Bilgic**

http://www.cs.iit.edu/~mbilgic

https://twitter.com/bilgicm

# Motivation

- So far, we talked about classification, where the target variable is discrete
- If the target is continuous, the task is called regression
- Examples
  - Recommendations (ratings)
  - Economics and finance (credit score, oil price, stock prices, consumption, etc.)
  - Weather forecasting (temperature, humidity, wind speed, etc.)
  - and more …

# Representation

- $x$: the input vector, the object
- $r$: the true value of the regression
- $f(x)$: the true underlying function
- $g(x)$: the estimated underlying function
- $\epsilon$: the noise
- $p(r|x)$: the conditional distribution of $r$ and $x$
- $D$: the dataset that consists of $\langle x, r \rangle$ pairs

# Regression Function

- $r = f(x) + \epsilon$
- $\epsilon$ is the noise (i.e., what the model cannot capture)
- Noise can exist due to several reasons
  - The input variables are insufficient to capture everything there is to capture
    - similar to uncertainty and probabilistic reasoning
  - Noisy sensors
- $\epsilon$ is typically assumed to be a zero mean Gaussian with constant variance $\sigma^2$
  - $\epsilon \sim \mathcal{N}(0, \sigma^2)$

# How To Learn $f(x)$

- One typical approach is to
  - Assume a parametric form
  - Formulate an objective function
  - Maximize or minimize it

# THUMBTACK TOSSES

- Imagine we have a thumbtack
- Flip it, and it comes as heads or tails

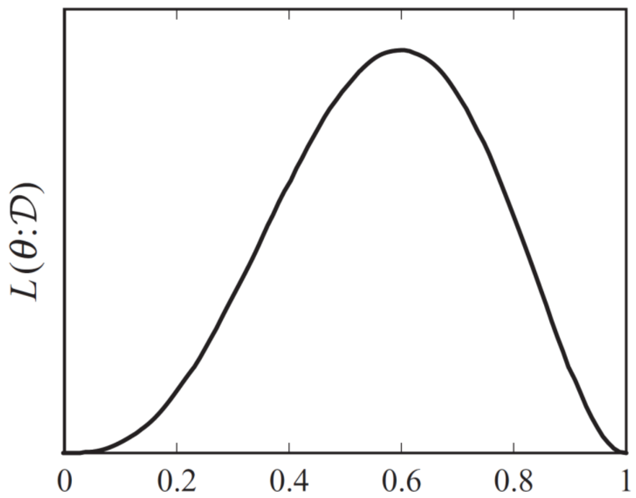

- P(Heads) = $\theta$, P(Tails) = 1- $\theta$
- Assume we flip it 100 times and it comes head 30 times
- What is $\theta$?

# Thumbtack tosses

- Assume we have a set of thumbtack tosses
  - $\mathcal{D} = \{d_1, \ldots, d_{\mathcal{M}}\}$
- Also assume each toss, $d_i$, is IID
- We define a *hypothesis space* $\Theta$
  - $\Theta$ is the set of all parameters $\theta \in [0, 1]$
- We formulate an *objective function*
  - The objective function tells us how good a given hypothesis (in this case $\theta$) is

# LIKELIHOOD

- What is the probability, or *likelihood*, of seeing the sequence H, T, T, H, H?
  - $\theta * (1 - \theta) * (1 - \theta) * \theta * \theta = \theta^3 (1 - \theta)^2$




When is $L(\theta : \mathcal{D})$ maximum?

# Likelihood/Log-likelihood

- Number of heads = h, number of tails = t
- Likelihood: $L(\theta:\mathcal{D}) = \theta^{h}(1-\theta)^{t}$
- Log-likelihood: $l(\theta:\mathcal{D}) = h\ln\theta + t\ln(1-\theta)$
- Find $\theta$ that maximizes the log-likelihood
- Take derivate of $l(\theta:\mathcal{D})$ with respect to $\theta$ and set it to zero

# Maximize Conditional Log-Likelihood For Regression

- Assuming $\epsilon \sim \mathcal{N}(0, \sigma^2)$, then
  - $p(r|x) \sim \mathcal{N}(g(x|w), \sigma^2)$
- Given the dataset $D$ that has $N$ instances and the parameter vector $w$, the conditional log-likelihood
  - $\text{CLL} = \sum_{i=1}^{N} \ln\left(p\left(r^{(i)}|x^{(i)}\right)\right)$
  - $\text{CLL} = \sum_{i=1}^{N} \ln\left(\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{\left(r^{(i)} - g(x^{(i)}|w)\right)^2}{2\sigma^2}}\right)$
  - $\text{CLL} = -N \ln\left(\sqrt{2\pi}\sigma\right) - \frac{1}{2\sigma^2} \sum_{i=1}^{N} \left(r^{(i)} - g(x^{(i)}|w)\right)^2$

# Maximize CLL = Minimize Squared Loss

- $\underset{\theta}{\operatorname{argmax}}\, CLL =$
- $\underset{w}{\operatorname{argmax}} -N\ln\left(\sqrt{2\pi}\sigma\right) - \frac{1}{2\sigma^2}\sum_{i=1}^{N}\left(r^{(i)} - g(x^{(i)}|w)\right)^2$
- $\underset{w}{\operatorname{argmin}} \sum_{i=1}^{N}\left(r^{(i)} - g(x^{(i)}|w)\right)^2$

# $g(x|w)$

- So far, we did not specify what $g(x|w)$ is
- One popular approach is to assume a linear function
- Assume in each instance $x$ has $k$ features
  - $x = \langle x_1, x_2, \cdots, x_k \rangle$
- Then, a polynomial of degree one linear regression is
  - $g(x|w) = w_0 + \sum_{i=0}^{k} w_i x_i$

## Gradient of the Squared Loss

- See OneNote

# Polynomial Regression – Arbitrary Degree

- Simply change the input representation by adding new features that correspond to powers and products
- See http://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.PolynomialFeatures.html#sklearn.preprocessing.PolynomialFeatures

# Regularization

- $L_2$ regularization
  - Minimize squared-loss + $L_2$ penalty
  - Also called Ridge regression
- $L_1$ regularization
  - Minimize squared-loss + $L_1$ penalty
  - Also called Lasso regression
- See OneNote for derivation

# RSE

- $RSE = \frac{\sum(r-g)^2}{\sum(r-\bar{r})^2}$
- RSE = closer to 1, if our prediction is as good/bad as predicting the mean all the time
- RSE = closer to 0 means we have a better fit
- Coefficient of determination
  - $R^2 = 1 - RSE$

# Transparency

- Generate a synthetic dataset using Hugin
- Analyze the weights with
  - No regularization
  - L2 regularization
  - L1 regularization
- How do the following impact weights?
  - Regularization
  - Feature scaling
  - Feature correlation

# Other Regression Approaches

- There are other approaches besides linear regression, such as
  - Decision tree regression
  - Support vector regression

# SCIKIT-LEARN

- Least squares
  - http://scikit-learn.org/stable/modules/linear_model.html#ordinary-least-squares
  - http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LinearRegression.html#sklearn.linear_model.LinearRegression
- Ridge
  - http://scikit-learn.org/stable/modules/linear_model.html#ridge-regression
  - http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Ridge.html#sklearn.linear_model.Ridge
- Lasso
  - http://scikit-learn.org/stable/modules/linear_model.html#lasso
  - http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Lasso.html#sklearn.linear_model.Lasso