Introduction to Machine Learning (by Implementation)

Lecture 3: Regression

Ian J. Watson

University of Seoul

University of Seoul Graduate Course 2019

KRF KOREA RESEARCH FELLOWSHIP
해외우수신진연구자 유치사업

Regression

- Regression is one of the major tasks in machine learning
- Idea: given some pieces of data, can you predict some dependent variables
- Examples
  - Can you predict the sale price of a house given location, size, no. of rooms, etc.
  - Given response of a calorimeter (measured in ADC counts), can you find the energy of the incoming particle
- Often the question is not so much to predict a quantity, but ask if there's a correlation between variables
  - Ex: Does the number of years in school impact your future salary
  - More concretely: how much of the variance in salary is explained by years of education
- Much of the statistics in regression is to say how significant the correlation between the variables is
- We'll contain ourselves mostly to the question of prediction, in particular *parametric regression*

Parametric Regression

- We have some variables $\mathbf{x}$ that represent our *measurement*, and we want to predict some y based on that
- In parametric regression, we build a *model*, a function $f(\mathbf{x}; \boldsymbol{\theta})$ which depends on the measurement variables and a set of *parameters* $\boldsymbol{\theta}$, which will *fit* or *train* to some known data sample
  - I.e. we have some known sample of $\mathbf{x}_i \rightarrow y_i$ which we will use to fix the parameters of the model
  - This is a *supervised learning* problem
- Example, we may have a sample $x_i \rightarrow y_i$, which we think can be modeled by a Gaussian, $f(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma}}$
  - The best estimate for μ will be the sample mean
  - The best estimate for σ^2 will be the sample variance
  - The above can be easily derived (we did in our stats class last semester)

Simple Linear Regression

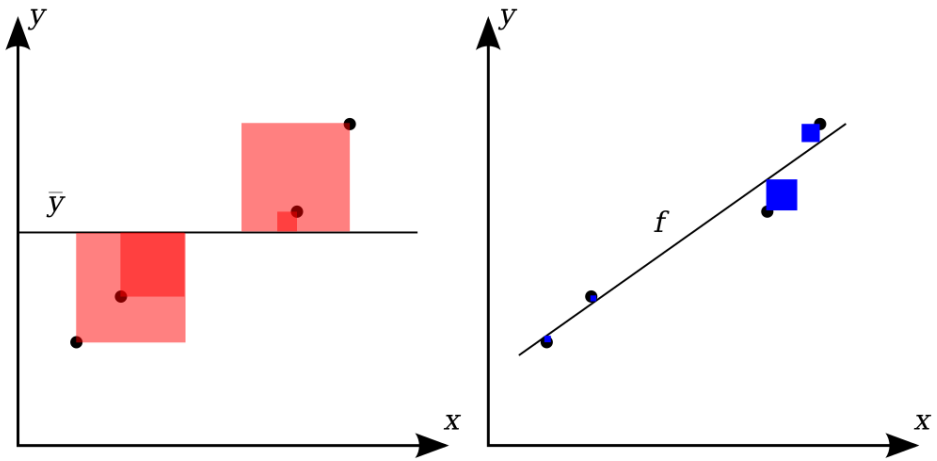
- Let's start with a basic model, simple linear regression
- You posit a linear relation between a single input variable x, and the output, y
  - $y_i = \beta x_i + \alpha + \epsilon_i$
- Given a simple model, it (usually) won't be able to explain all the variance, ϵ_i represents the *residual*
  - Residual: error term for factors not accounted for by the model
- For x_i with unknown y, we would use the model to make a prediction: $y_i \approx \beta x_i + \alpha$
- The best model will minimize the size of the residual
  - Equivalently, minimize the sum of the square of the residuals $\sum_i \epsilon_i^2$
  - Why SSR?
    - Easier to do calculus on than absolute value
    - Equivalent to maximum likelihood estimate assuming the residuals are Gaussian distributed with known variance

Simple Linear Regression (continued)

- The function which we want to minimize in machine learning is commonly called the *loss function* (assume $i = 1, 2 \ldots n$)
  - $L = \sum_i \epsilon_i^2 = \sum_i (y_i - \beta x_i - \alpha)^2$, is the *residual sum of squares*
- To find the minimum, need to find
  - $\frac{\partial L}{\partial \alpha} = \sum_i -2(y_i - \beta x_i - \alpha) = 0$
  - $\frac{\partial L}{\partial \beta} = \sum_i -2x_i(y_i - \beta x_i - \alpha) = 0$
  - Some algebra ...
  - $\alpha = \frac{1}{n}\sum_i y_i - \beta \frac{1}{n}\sum_i x_i = \langle y \rangle - \beta \langle x \rangle$
  - $\beta = \frac{\langle xy \rangle - \langle x \rangle \langle y \rangle}{\langle x^2 \rangle - \langle x \rangle^2} = \frac{cov(x,y)}{var(x)} = corr(x,y)\frac{\sigma_y}{\sigma_x}$
- The formulae for α and β give the exact least squares fit for the regression coefficients
  - In this simple case, we can derive exactly, next week, we will go to a more complicated case, which requires our minimizer from last week

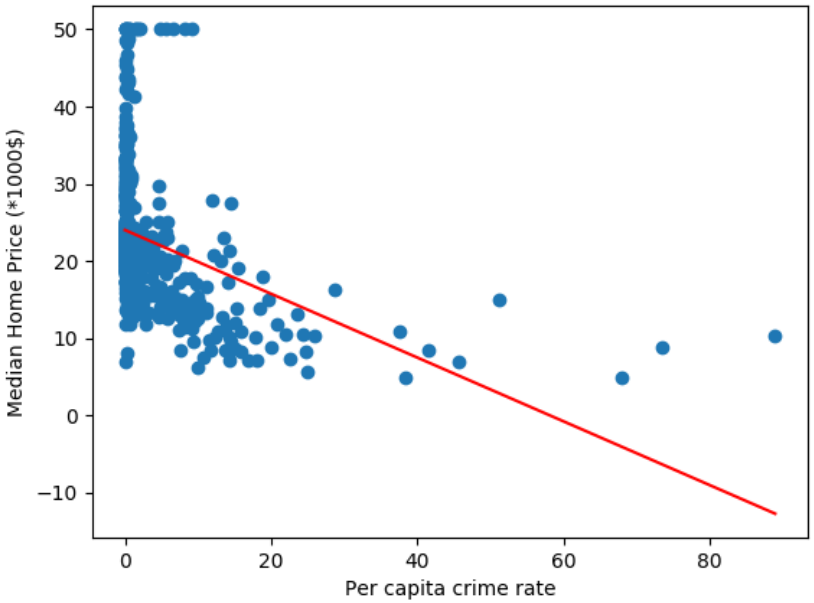
Reminder of some statistics terms and my terminology:
$\langle g \rangle = \frac{1}{n}\sum_i g(x_i, y_i)$ is the average value of some function g of the data
$\langle x \rangle = \frac{1}{n}\sum_i x_i$ is the mean. $var(x) = \langle x^2 \rangle - \langle x \rangle^2$ the variance, $\sigma_x = \sqrt{var(x)}$.
$cov(x, y) = \langle xy \rangle - \langle x \rangle \langle y \rangle$ is the covariance. $corr(x, y) = \frac{cov(x,y)}{\sigma_x \sigma_y} \in [0, 1]$ is the correlation.

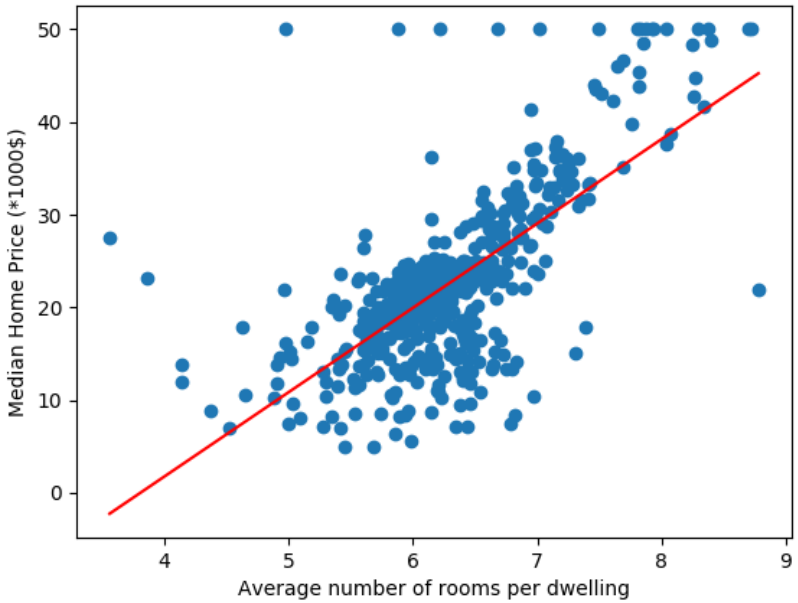
R^2, the coefficient of determination



- We can ask for a measure of how well a model f_i "explains" the dependent variable y_i
- The *coefficient of determination* or R^2 measures the fraction of the total variation captured by the model
- $R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$, where
  - $SS_{tot} = \sum_i (y_i - \langle y \rangle)^2$, the total sum of squares, or total variance
  - $SS_{res} = \sum_i (y_i - f_i)^2 = \sum_i \epsilon_i^2$, the residual sum of squares
- The $\frac{SS_{res}}{SS_{tot}}$ is also called *the fraction of variance unexplained*
- $R^2 \in [0, 1]$,
  - If $R^2 = 0$ then the model is no better than simply using the y_i mean
  - If $R^2 = 1$ then the residuals are all 0, no variance left in data

Example plots of the boston dataset





- We will look at the "Boston Housing Dataset" taken from census data
- Gives several variables for a housing tract (block) and the dependent variable (variable to be described/predicted) of the median value of owner-occupied houses
- Plots for the 0th column (left), and the 5th column (right) of data
- You should make plots and do a linear regression for all the columns

Harrison and Rubinfeld, Journal Of Environmental Economics And Management 5, 81-102 (1978)

Exercises

- Make scatter plots (or ROOT TGraph) for each of the variables in the boston dataset
  - Write them to a file "<n>.png" where <n> is the 0-indexed variable number
- Write a function `predict(alpha, beta, x_i)` which predicts the output from a single datapoint, given the alpha, beta
- Write a function to perform a simple linear regression `linear_regression`
  - `linear_regression_least_squares(x: List[float], y: List[float]) -> (float, float)` takes the data, the known values, and returns (α, β) from the least squares
    - Perhaps a `mean(x: List[float])` function would be useful? What about a `variance(x: List[float])` and `correlation(x: List[float], y: List[float])` function?
- Write a function which finds the coefficient of determination R^2 `r_squared`
  - `r_squared(alpha: float, beta: float, x: List[float], y: List[float])`
  - Where $\beta x_i + \alpha = f_i$ the prediction for y_i

Exercises (cont'd)

- For the boston dataset, run the linear regression and find R^2 for each of the columns (i.e. treat each column independently)
  - Write out into a file `results.txt`, in order of the columns, the alpha, beta and R2 values you found, separated by a comma
  - Once for each column on a separate line with no other information
  - Do the values match your intuition from the plots?
- Commit the python code, the png's and results.txt
  - There is no pytest code this week, you'll have to use your judgement/write your own tests!
  - A simple test could be to make a small dataset that you know the answer to, and test that the code gives you the answer
  - Eg (x,y) = [(0,0), (1,1)]. What should be alpha and beta?

(Very) Basic Matplotlib

If the import fails, run `pip install matplotlib` in your virtualenv

```
import matplotlib.pyplot as plt

# If you have x and y, two lists of the same length, can plot
# the (x, y) scatterplot as:
plt.clf()  # clear the figure (remove any previous plots)
plt.scatter(x, y)  # Make the plot
plt.savefig("xy.png")  # Write the plot to the file xy.png

# To draw a line from (x1, y1) to (x2, y2), use plot
plt.plot([x1, x2], [y1, y2], color='r') # Draw line in red
plt.savefig("xy.png")  # Write the plot to the file xy.png

# without a "plt.clf()", the two plots are overlaid
```