Lesson Plan 4/5

Thursday, April 5, 2018 11:31 AM

Admin
- Project 2 due on Friday
- HW4 due on Tuesday
- Grades for Project 1 posted
- HW3 grading almost done

HDSC:
- Example wrangling with R and Python: https://www.superdatascience.com/wrangling-[illegible]data-r-python/

Project 2 Questions?
- Added note re. dates to project description

HW4
- Questions?

Data Analysis with Geometry
- Preliminaries
- Geometry and distances
- KNN classifier
- Some vector algebra

Linear Regression
- Formulation
- Interpretation

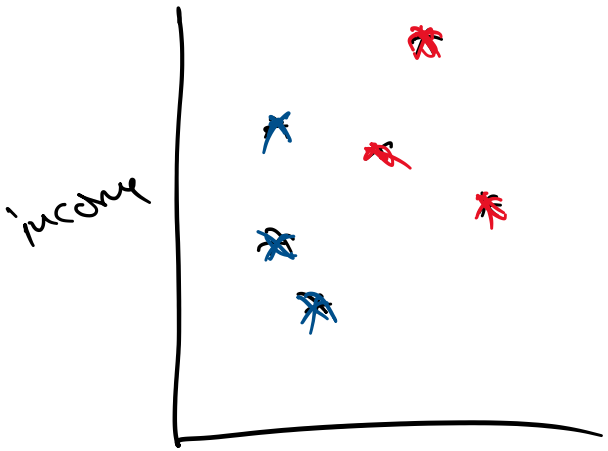
X: predictors

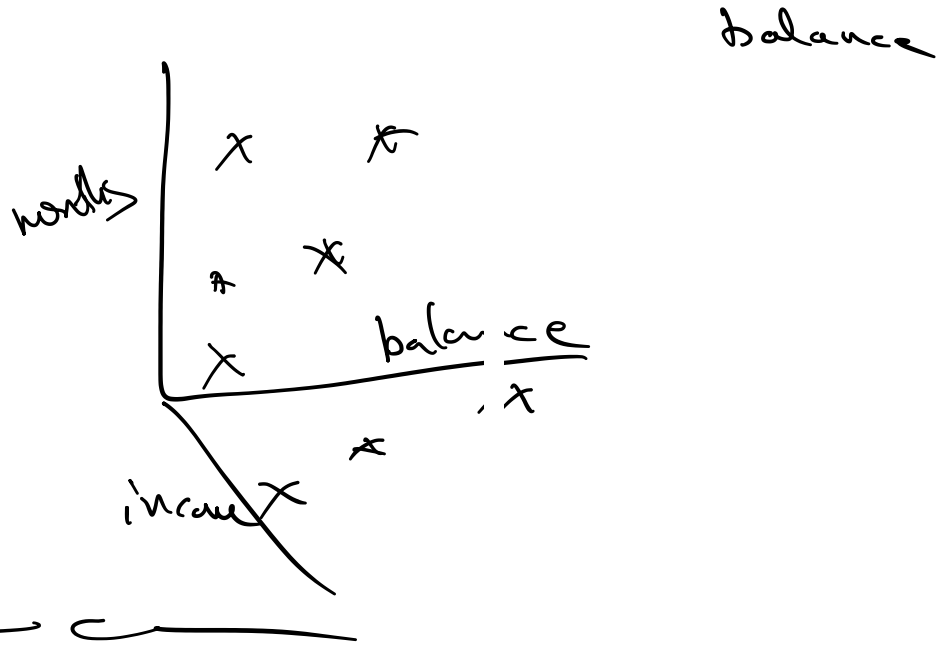
G: categorical outcome

Represent entities as points in Euclidean Space

Entities are represented as $\langle x_1, x_2, \ldots, x_p, y \rangle$

$\in \mathbb{R}^{p+1}$





Distances

- Euclidean Distance

$$d(x_1, x_2) = \sqrt{\sum_{i=1}^{p} (x_{1i} - x_{2i})^2}$$

Inductive Bias

→ Examples that are "close" have similar outcomes

$$P(Y = y \mid X = x)$$

points that are "close"

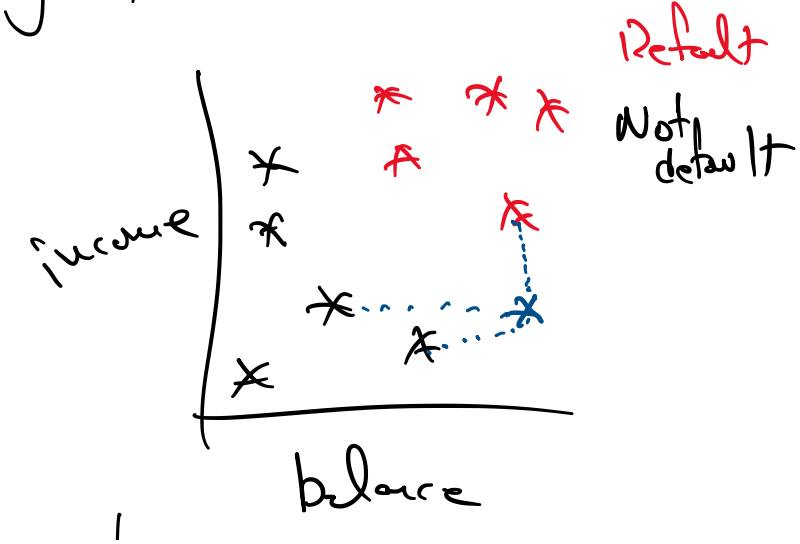
(1) Categorical attributes
    ↓ transformed into
    numeric

(2) Scaling & centering important

K- nearest neighbor algorithm

Given training set

$\{\langle x_{11}, \ldots, x_{1p}, y_1\rangle,$
$\langle x_{21}, \ldots, x_{2p}, y_2\rangle$
$\vdots$
$\{\langle x_{n1}, \ldots, x_{np}, y_n\rangle\}$



Generate predictions for new examples $\langle x_{t1}, \ldots, x_{tp} \rangle = x_t$.

→ Compute distance between x_t & each point in training set

→ Find K nearest points

→ Compute majority class

→ Return majority class

Why choose K?

Case 1) $K = n$

All predictions are the same (underfitting)

Case 2) $K = 1$

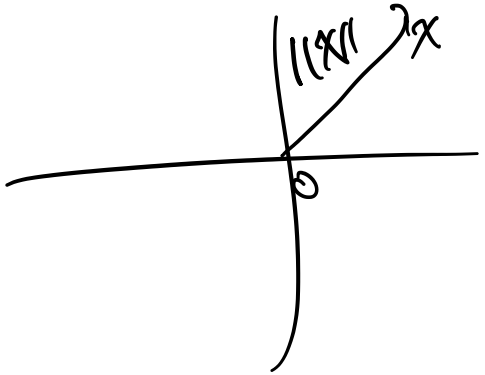
"Unstable" (overfitting)

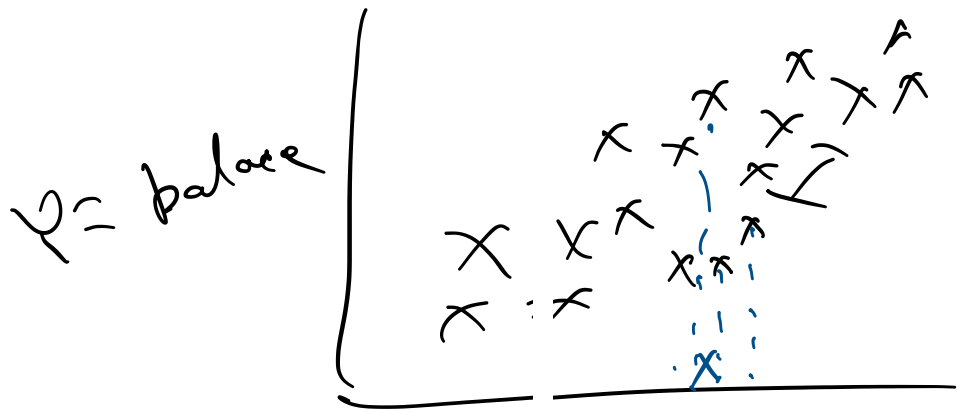
K: hyper-parameter

interested over vs. underfitting

generalization: ability to accurately predict outcomes for unseen data

$$\|x\| = \sqrt{\sum_{j=1}^{p} (x_j - 0)^2}$$





Now I have

$y_1, y_2, y_3, \ldots, y_k$ for k nearest neighbors

$\bar{y}$ minimizes $\min \sum_{i=1}^{k} (y_i - \mu)^2$

Assume

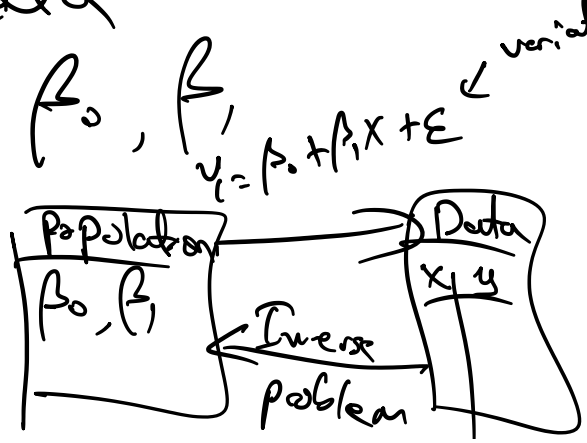
$$Y = \beta_0 + \beta_1 X$$

Linear regression model

Estimation problem:

- Given training data
- learn parameters β_0, β_1

$$\min \sum (y_i - (\beta_0 + \beta_1 x_i))^2$$



β_0, β_1 : $(\lambda_i \quad x_0 \pi_1 \hat{\ } i)$

=

Model

⟶ Closed-form solution for $p = 1$

⟶ Algorithms to minimize this function