



Feature Selection & Prediction in Public Health

Oliver Bradley, Julianne Higgins, Lesley Tan



Chosen Lab + Extension

- Netflix lab
 - Feature selection
 - Making predictive models
- Applying these concepts to a public health problem

→ **Feature selection** is just as important, if not more important, than **model selection**.

Research question:

- What **method of feature selection** can reach accurate predictions with **low computational cost**?



Methods + Data Overview

Data Cleaning

Iterations

Randomization

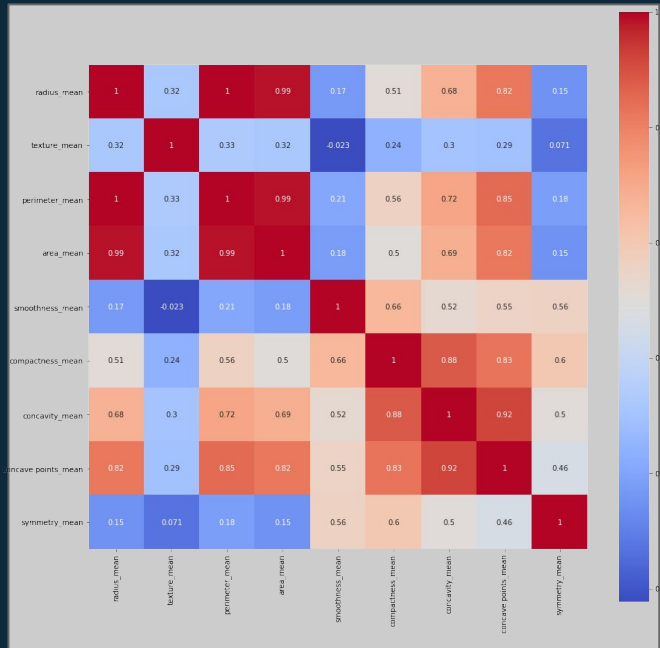
Dataset overview

- **Wisconsin Breast Cancer Diagnostic Dataset** from UCI Machine Learning Repository
- 12 variables (ID, diagnosis and 10 predictors), 569 observations
- Response Variable
 - **Diagnosis** (M = Malignant, B = Benign)
- Predictors
 - radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry, and fractal points

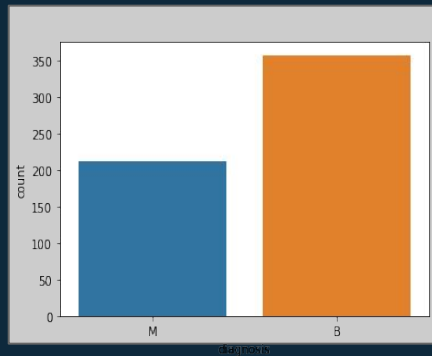


Data Exploration and Cleaning

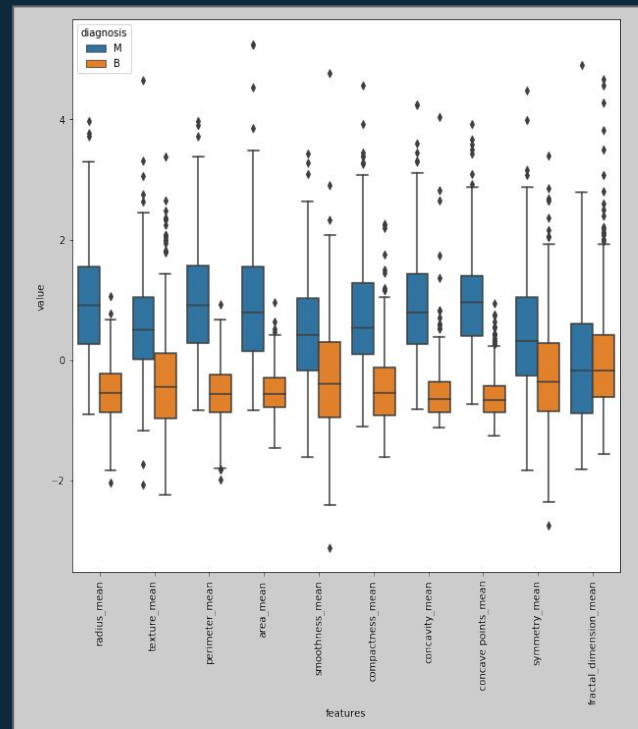
Heat Map of Predictors



Distribution of Response Variable



Distribution of our Predictors






Logistic Regression with 5 variables

Head of our “clean” data:

	radius_mean	texture_mean	smoothness_mean	concavity_mean	symmetry_mean
0	17.99	10.38	0.11840	0.3001	0.2419
1	20.57	17.77	0.08474	0.0869	0.1812
2	19.69	21.25	0.10960	0.1974	0.2069

- Fit logistic regression to all 5 variables
 - Cross validation =10, Accuracy score = 0.8966
- Will we find a better result when fitting log reg to different combinations of features?
 - Tested all 31 combinations of features
 - Resulted in same model which includes all 5 features


$$\sum_{i=1}^5 \binom{5}{i} = 31$$



Logistic Regression with interaction variables

- How will the feature selection change if we add interaction terms?
- There are 10 pairwise interaction terms out of 5 predictors

$$\binom{5}{2}$$

- Iterated through every combination of the 15 predictors

$$\sum_{i=1}^{15} \binom{15}{i} = \underline{32\,767}$$

- Model with all 15 predictors
 - Accuracy score of 94.05%
 - 2-hour run time



Overfitting?

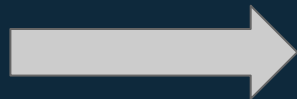




Random Search for Feature Selection

- Reminiscent of Monte Carlo Simulations

- A **better approach** is to use **randomization** to select features for a predictive model



- Reduces computational load
- Provides more balanced model

- Define N amount of iterations
- Randomly picks number of variables to use, then randomly selects predictors
- Fits and cross validates model, records accuracy and model in dictionary
- Sort dictionary by highest accuracy






Results + Conclusion

We found random search to be an effective feature selection method, and allowed us to control computational load.

- Now we have a method to search for viable models
- Can also scan for common predictors which appear in higher performing models

Future Research:

- Can apply random search to different models
 - Can include higher dimensionality and more variables in model as the computational ceiling is lower in random search
- 



Thanks!

Any questions?

