MMAI 5040 – Winter 2022
Business Application of AI 1

Session 3: Regression Frameworks for Machine Learning

January 24, 2021

Divinus Oppong-Tawiah, PhD.

Today's Class …

1. Announcements & Recap

2. Case Discussion: Data Science at the Warriors

3. Regression Frameworks in ML
    - Multiple Linear Regression
    - Logistic Regression
    - Feature Selection
    - Business Application

Announcements:

- Course Project
  - Groups will set up in Canvas and announced via email
  - Resources for project (e.g., data sources will be posted).
  - Sign up sheet for group appointments at office hours (if you want to discuss your initial ideas / challenges).

- Tutorial materials (script, data, videos) will be posted after class

- Assignment 1 will be posted today. E-mail announcement will follow.

Table 1. A Summary of Data Pre-processing Tasks and Potential Methods (Recap Last Week).

Main Task	Subtasks	Popular Methods
Data consolidation	Access and collect the data. Select and filter the data Integrate and unify the data	SQL queries, software agents, Web services. Domain expertise, SQL queries, statistical tests. SQL queries, domain expertise, ontology-driven data mapping.
Data cleaning	Handle missing values in the data	Fill in missing values (imputations) with most appropriate values (mean, median, min/max, mode, etc.); recode the missing values with a constant such as "ML"; remove the record of the missing value; do nothing.
	Identify and reduce noise in the data	Identify the outliers in data with simple statistical techniques (such as averages and standard deviations) or with cluster analysis; once identified, either remove the outliers or smooth them by using binning, regression, or simple averages.
	Find and eliminate erroneous data	Identify the erroneous values in data (other than outliers), such as odd values, inconsistent class labels, odd distributions; once identified, use domain expertise to correct the values or remove the records holding the erroneous values.
Data transformation	Normalize the data	Reduce the range of values in each numerically valued variable to a standard range (e.g., 0 to 1 or −1 to +1) by using a variety of normalization or scaling techniques.
	Discretize or aggregate the data	If needed, convert the numeric variables into discrete representations using range- or frequency-based binning techniques; for categorical variables, reduce the number of values by applying proper concept hierarchies.
	Construct new attributes	Derive new and more informative variables from the existing ones using a wide range of mathematical functions (as simple as addition and multiplication or as complex as a hybrid combination of log transformations).
Data reduction	Reduce number of attributes	Use principal component analysis, independent component analysis, chi-square testing, correlation analysis, and decision tree induction.
	Reduce number of records	Perform random sampling, stratified sampling, expert-knowledge-driven purposeful sampling.
	Balance skewed data	Oversample the less represented or undersample the more represented classes.

CASE DISCUSSION:
DATA SCIENCE AT THE WARRIORS

Data Science at the Warriors

Case Discussion Questions

1. Case Summary: What is this case about? Who are the protagonists? What are their roles?

2. What is the notion of 'safe deployment'? (Hint: When is a model ready for live deployment?)

3. Skillset: Consider the definition of Data Science on page 2 and the required skills and responsibilities in Exhibit 2. Do you see any advantages your MMAI training brings over a pure data scientist?

4. Impact and feasibility: What other questions would you ask in the Project Feasibility Check List? (Exhibit 3).

5. Data limitations and model accuracy: What are the team's major concerns here? Are they valid? Why or why not?

6. Consider the project process flow: EDA → Statistical Inference → Prediction → Causal Inference. Can you explain the different goals of each stage? Are all of them necessary? Why or why not?

7. Decision time: Should the team test the model's real-world efficacy? Is the experimental design appropriate? What other methods could help (faster, cheaper, reliable, consistent, robust, etc..)?

Break: 10 minutes

MULTIPLE LINEAR REGRESSION

Multiple Linear Regression

1. The Purpose of Linear Regression
2. The Regression Modeling Process
3. Business Application Issues
4. Linear Regression in Python (Tutorial)
5. Examples (Tutorial)

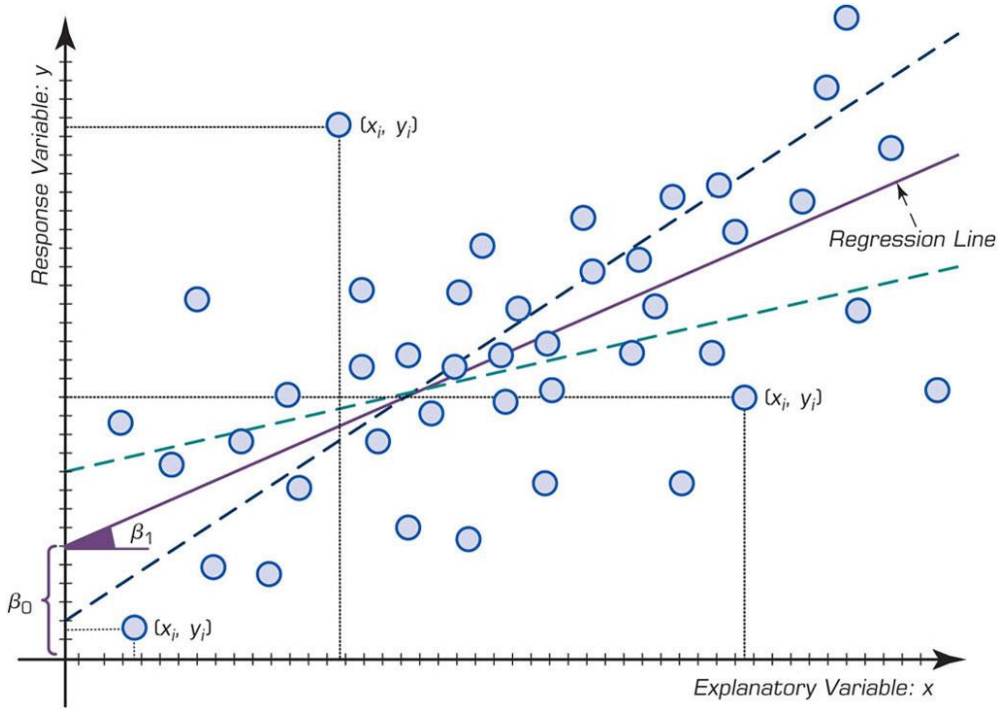
Purpose of Linear Regression

- Descriptive & Explanatory Modeling
  - Quantify the average effect of inputs on an outcome with causal structure unknown (Descriptive) or known (Explanatory)
  - Familiar use of regression in statistics
  - Model Goal: Fit the data well and understand the contribution of explanatory variables to the model
  - "goodness-of-fit": R-squared, residual analysis, p-values

- Predictive Modeling
  - Predict the outcome value for new records given their input values
  - Classic ML context
  - Model Goal: Optimize predictive accuracy
  - Train model on training data
  - Assess performance on validation (hold-out) data
  - Explaining role of predictors is not primary purpose (but useful)

Multiple Linear Regression

1. The Purpose of Linear Regression
2. The Regression Modeling Process
3. Business Application Issues
4. Linear Regression in Python (Tutorial)
5. Examples (Tutorial)



The Regression Modeling Process

- We assume a linear relationship between predictors and outcome:

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p + \varepsilon$$

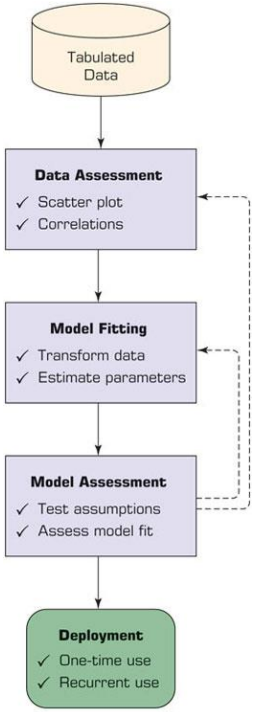
coefficients: β_0, β_1, β_2, β_p

outcome: Y | constant: β_0 | predictors: x_1, x_2, x_p | error (noise): ε

- Goal: OLS regression: minimize the squared errors between observations and the regression line

- Other that assumptions may /may not apply:
  - Explanatory vs Predictive goal

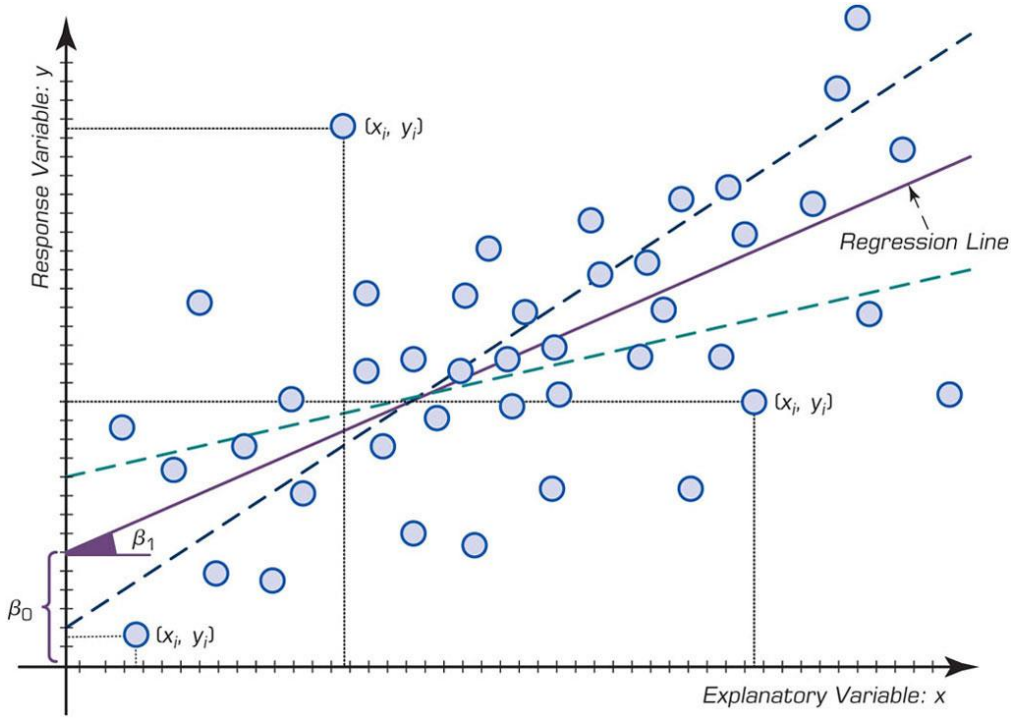
- How do we know if the model is good enough?
  - R-square
  - p values
  - Error measures (for prediction tasks)



Some OLS estimation assumptions:
1. Normally distributed errors
2. Linear predictors
3. I.I.D records
4. Constant variance
5. Multicollinearity

Multiple Linear Regression

1. The Purpose of Linear Regression
2. The Regression Modeling Process
3. Business Application Issues
4. Linear Regression in Python (Tutorial)
5. Examples (Tutorial)



Business Application Issues

- Multicollinearity
- Inference
  - Correlation vs. Causation
  - Variable selection
- Interpretation
  - Coefficient value
  - Statistical significance
    - T-test
    - F-test

Linear Regression in Python

- Statistics Perspective
  - statsmodels package
  - obtain stats-related results (t-value, p-value, etc.)
- Machine Learning Perspective
  - sklearn package
  - compatible result from standard sklearn functions (cross-validation, MSE calculation, etc.)

Example

Prices of Toyota Corolla
ToyotaCorolla.xls

Goal: predict prices of used Toyota Corollas based on their specification

Data: Prices of 1000 used Toyota Corollas, with their specification information

Output of the Regression Model

Partial Output

	Predictor	coefficient
0	Age_08_04	-140.748761
1	KM	- 0.017840
2	HP	36.103419
3	Met_Color	84.281830
4	Automatic	416.781954
5	CC	0.017737
6	Doors	-50.657863
7	Quarterly_Tax	13.625325
8	Weight	13.038711
9	Fuel_Type_Diesel	1066.464681
10	Fuel_Type_Petrol	2310.249543

Make the Predictions for the Validation Data (and show some residuals)

	Predicted	Actual	Residual
507	10607.3339	11500	892.6660
818	9272.7057	8950	-322.7057
452	10617.9478	11450	832.0521
368	13600.3962	11450	-2150.3962
242	12396.6946	11950	-446.6946
929	9496.4982	9995	498.5017
262	12480.0632	13500	1019.9367

Accuracy Metrics on the Training Data (Traditional Metrics) :

Regression statistics	
Mean Error (ME)	: 0.0000
Root Mean Squared Error (RMSE)	: 1400.5823
Mean Absolute Error (MAE)	: 1046.9072
Mean Percentage Error (MPE)	: -1.0223
Mean Absolute Percentage Error (MAPE)	: 9.2994

How Well did the Model Do With the Validation Data?

Accuracy Metrics for the Regression Model:

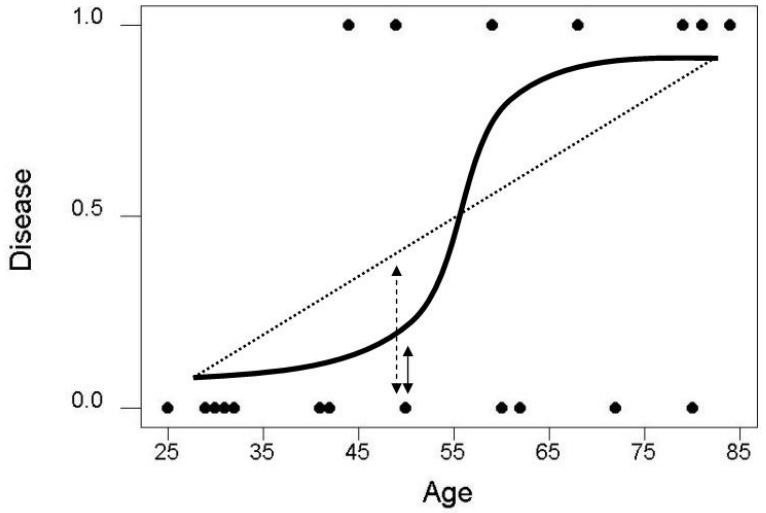
Regression statistics	
Mean Error (ME)	: 103.6803
Root Mean Squared Error (RMSE)	: 1312.8523
Mean Absolute Error (MAE)	: 1017.5972
Mean Percentage Error (MPE)	: -0.2633
Mean Absolute Percentage Error (MAPE)	: 9.0111

LOGISTIC REGRESSION

Logistic Regression

1. The Purpose of Logistic Regression
2. Logistic Regression Modeling
3. Business Application Issues
4. Example in Python Example (Tutorial)

Plot A: Effect of Age on Disease



Purpose of Logistic Regression

- Extends idea of linear regression to situation where outcome variable is categorical. Why is linear regression not appropriate here?
  - Binary data typically does not have a normal distribution
  - Predicted value of the dependent variable can be beyond 0 and 1
  - Probabilities are often not linear (e.g., "U" shapes)

- Plot A shows least squares regression line (straight) and logistic regression line (curved) for a regression of disease on age.
  - What does each model assume about the effect?
    - Linear versus non-linear relationship
  - What can you say about the estimation error for patient 11?
    - High vs low estimation error

Logistic Regression

1. The Purpose of Logistic Regression
2. Logistic Regression Modeling
3. Business Application Issues
4. Example in Python Example (Tutorial)

$$p = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_q x_q \quad (1)$$

$$p = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_q x_q)}} \quad (2)$$

$$\text{Odds} = \frac{p}{1 + p} \quad (3)$$

$$p = \frac{\text{Odds}}{1 + \text{Odds}} \quad (4)$$

$$\text{Odds} = e^{(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_q x_q)} \quad (5)$$

$$\log(\text{Odds}) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_q x_q \quad (6)$$

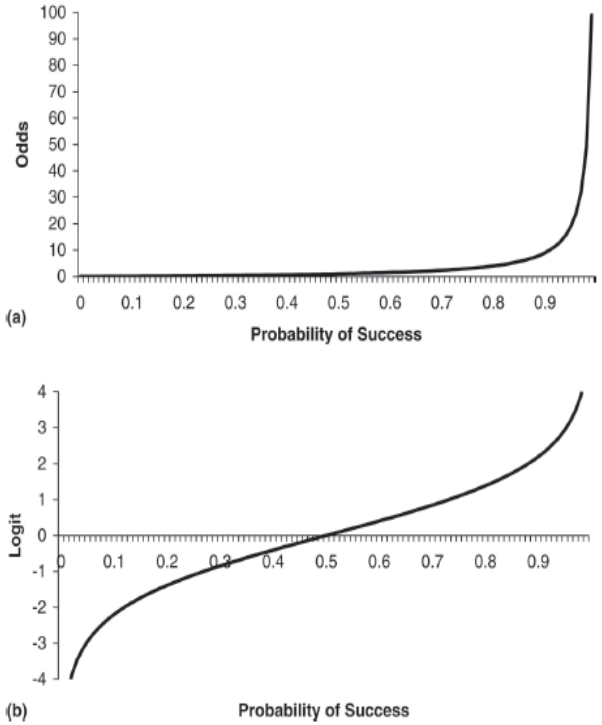
Logistic Regression Modeling: The Logit Function

- **Goal:** Find a function of the predictor variables that relates them to a 0/1 outcome (i.e., we focus on binary classification here).
- Instead of Y as outcome variable, we use a function of Y called the *logit*
- Logit can be modeled as a linear function of the predictors:
  - if p = probability of belonging to class 1, relate p to predictors with a function that guarantees $0 \leq p \leq 1$
  - Standard linear function (1) does not (see previous slide)
  - The fix: use logistic response function (2) with curvilinear shape
  - The odds of an event are defined as (3):
  - Or, given the odds of an event, the probability of the event can be computed by (4):
  - Relate the odds to the predictors by substituting (2) into (4)
  - We obtain the logit by taking the log on both sides of (5): i.e., log(odds) = logit (6)

Logistic Regression

1. The Purpose of Logistic Regression
2. Logistic Regression Modeling
3. Business Application Issues
4. Example in Python Example (Tutorial)

Odds & Logit as a function of p



Logistic Regression Modeling: The Logit Function

- So, the logit is a linear function of predictors x1, x2, …
  - Takes values from -infinity to +infinity
  - β's are derived through *maximum likelihood estimation*
  - The logit can be mapped back to a probability, which, in turn, can be mapped to a class

Business Application Issues

- Feature selection (especially in *profiling* tasks)
- Cutoff for converting to a classification
  - If estimated prob. > cutoff, classify as "1"
  - How to determine cutoff?
  - 0.50 is popular initial choice
  - Additional considerations (recall from Class 2)
    - Maximize classification accuracy
    - Maximize sensitivity (subject to min. level of specificity)
    - Minimize false positives (subject to max. false negative rate)
    - Minimize expected cost of misclassification (specify costs)

Example

Personal Loan Offer
(UniversalBank.csv)

Outcome variable:
accept bank loan (0/1)

Predictors:
- Demographic info,
- info about bank relationship

Results:

```
Coefficients for logit

intercept   -12.618955

Age          -0.032549
Experience    0.03416
Income        0.058824
Family        0.614095
CCAvg         0.240534
Mortgage      0.001012
Sec_Acct     -1.026191
CD_Acct       3.647933
Online       -0.677862
CreditCard   -0.95598
Edu_Graduate  4.192204
Edu_Pro       4.341697

AIC -709.1524769205962
```

Converting from logit to probabilities

```
logit_reg_pred = logit_reg.predict(valid_X)
logit_reg_proba = logit_reg.predict_proba(valid_X)
logit_result = pd.DataFrame({'actual': valid_y,
   'p(0)': [p[0] for p in logit_reg_proba],
   'p(1)': [p[1] for p in logit_reg_proba],
   'predicted': logit_reg_pred })

# display four different cases
interestingCases = [2764, 932, 2721, 702]
print(logit_result.loc[interestingCases])
```

```
      actual   p(0)    p(1)    predicted
2764     0     0.976   0.024    0
932      0     0.335   0.665    1
2721     1     0.032   0.968    1
702      1     0.986   0.014    0
```

Interpreting Odds, Probability

- For predictive classification, we typically use probability with a cutoff value
- For explanatory purposes, odds have a useful interpretation:
  - If we increase x_1 by one unit, holding x_2, x_3 … x_q constant,
  - then b_1 is the factor by which the odds of belonging to class 1 increases

FEATURE SELECTION

Feature Selection

1. General Concepts
2. Tools & Techniques
3. Feature Selection in Regression Frameworks
4. Example in Python Example (Tutorial)

General Concepts

- Feature Selection is a process to choose a subset of features (variables) to improve model performance
- Reduces model complexity
  - The number of correlated predictors can grow with big data, or when we create derived variables such as interaction terms to capture more complex relationships (danger of overfitting)
- Improves model efficiency
  - Removing irrelevant data reduces the cost of data collection, storage requirements, computational resources, etc.
- Improves model interpretability
  - high transparency and understanding of the data and the model

- Caveats:
  - reduces training accuracy but could increase validated accuracy
  - NOT same as feature extraction: Feature extraction transforms non-numerical input (e.g., text, image, etc.) into numerical features usable for machine learning
  - NOT same as dimension reduction: Dimension reduction (e.g., PCA) combines several features together as components.

Feature Selection

1. General Concepts
2. Tools & Techniques
3. Feature Selection in Regression Frameworks
4. Example in Python Example (Tutorial)

Tools

- Statistical tools for reducing model complexity:
  - Variable selection in regression analysis (e.g., forward, backward, step-wise, best subset)
  - Principal Component Analysis (PCA) (but reduces interpretability)
  - Factor Analysis groups predictors into fewer latent variables
- ML algorithms e.g.,
  - Decision tree because it only includes relevant features in the tree
  - Clustering can also group features that are highly correlated

Techniques

- Filter methods: filter the features based on certain criteria (no ML)
- Wrapper methods: Use some ML algorithms to recursively evaluate model performance on the set of features and select the best set:
  - *e.g., Recursive Feature Elimination* (works with all classifiers)
- Embedded methods: Use specifications of some ML algorithms to pick the most importance features;
  - e.g., Random Forest *('feature importance' score)*, Regularization (*shrinkage of regression coefficients*)

Feature Selection

1. General Concepts
2. Tools & Techniques
3. Feature Selection in Regression Frameworks
4. Examples in Python (Tutorial)

$$R^2_{adj} = 1 - \frac{n-1}{n-p-1}(1-R^2)$$

Penalty for number of predictors in exhaustive search

Comparing Methods (Tutorial)

(results same in this data, but it need not be so)

Variable	Forward	Backward	Stepwise	Exhaustive
Age_08_04	✔	✔	✔	✔
KM	✔	✔	✔	✔
HP	✔	✔	✔	✔
Met_Color				
Automatic	✔	✔	✔	✔
CC				
Doors				
Quarterly_Tax	✔	✔	✔	✔
Weight	✔	✔	✔	✔
Fuel_TypeDiesel	✔	✔	✔	✔
Fuel_TypePetrol	✔	✔	✔	✔

Feature Selection in Regression Frameworks

- Two main techniques: Variable selection & Regularization

- Variable Selection (Selecting subsets of predictors)
  - **Goal:** Find parsimonious model (the simplest model that performs sufficiently well, for both linear and logistic regression)
  - More robust model, higher predictive accuracy
  - We will assess predictive accuracy on validation data using
    - Exhaustive Search = "best subset"
      - Assess all possible subsets (single, pairs, triplets, etc..)
      - Judge by "adjusted R^2"
      - Computationally intensive, not feasible for big data
    - Partial Search Algorithms
      - Forward : Start with no predictors, add them one by one, stop when the addition is not statistically significant
      - Stepwise: Like forward search, but at each step, also consider dropping non-significant predictors
      - Backward: Start with all predictors, successively eliminate least useful predictors one by one, stop when all remaining predictors are statistically significant

Feature Selection

1. General Concepts
2. Tools & Techniques
3. Feature Selection in Regression Frameworks
4. Examples in Python (Tutorial)

Feature Selection in Regression Frameworks

- Two main techniques: Variable selection & Regularization

- Regularization (Shrinkage Models)
  - **Goal:** Rather than binary decisions on including variables, penalize coefficient magnitudes by shrinking them to reduce variance
  - Predictors with coefficients that shrink to zero are dropped
  - OLS minimizes sum of squared errors (residuals) – SSE
  - Shrinkage models minimize SSE subject to penalty being below specified threshold, e.g.:

    - LASSO (Least Absolute Shrinkage and Selection Operator) uses penalty *L1, the sum of absolute values for coefficients*

    - Ridge regression uses penalty *L2, the sum of squared coefficients*

    - Predictors are typically standardized

Break: 10 minutes

BUSINESS APPLICATION OF REGRESSION FRAMEWORKS: A PUBLIC HEALTH CASE

Reducing Heart Disease Risk in the Population

1. Problem
2. Ideal Solution
3. Constraints
4. Practical Solution
5. ML Task
6. Data
7. Model Results & Interpretation

Reducing Heart Disease Risk in the Population

- **Problem:** Heart disease has high mortality rate (1/1000), high incidence rate
- **Ideal solution:** Invite everyone for diagnostic test / medical advice
- **Constraints:** High cost, few doctors, need to treat other diseases, etc.
- **Practical solution:** Identify at least half of those at most risk in the next 5 years, yet target at most 5% of the population (i.e. invite only 1 in 20 with a high enough hit rate).
- **ML task:** Estimate the likelihood of developing heart disease in the next 5 years and use the model to invite those at most risk.
- **Data:** N = 2ML random sample with no sign of heart disease 5 years ago; 500,000 random split (validation set). Predictors = Age, gender, BMI, med history, BP, smoke, alcohol, etc.
  Outcome = Heart disease diagnosis within last 5 years [incl. 30,000 (6%) positive cases]
- **Regression Model Results & Interpretation:** (next)

Fig A. Model Output: A scorecard for predicting heart disease

Starting score (constant)	350		
Age (years)		**Gross annual income ($)**	
<23	-57	< $22,000	11
23 - 32	-26	$22,001 - $38,000	6
33 - 41	0	$38,001 - $60,000	0
42 - 48	7	$60,001 - $94,000	-3
49 - 57	15	$94,001 - $144,000	-5
58 - 64	24	>$144,000	-6
65 - 71	31		
>71	65	**Smoker ?**	
		Yes	37
BMI (weight in kg / {height in metres}^{2})		No	0
<19	2		
19 - 26	0	**Diabetic ?**	
27 - 29	8	Yes	21
30 - 32	14	No	0
>32	29		
		Cholesterol level (mg per decilitre of blood)	
Gender		Low (< 160 mg)	-2
Male	2	Normal (160 - 200 mg)	0
Female	-4	High (201 - 240 mg)	19
		Very high (>240 mg)	32
Alcohol consumption (units/week)			
0	4	**Blood pressure**	
1 - 12	0	Low (below 90/60)	3
13 - 24	5	Average (between 90/60 and 140/90)	0
25 - 48	10	High (above 140/90)	36

Interpreting Results:

Examine the scorecard for

1. Individual Scores (target prediction)
2. Transparency (White box)
3. External validation (expected / unexpected)
4. Parsimony and robustness
5. Score probabilities (or propensities)

Target: 45 y/o female, BMI =28, Alc = 6, Income = $50K,
Smokes, Non-diabetic, Normal cholest., Low BP

Score = 350 + 7 + 8 – 4 + 0 + 0 + 37 + 0 + 0 + 3
= 401

Fig A. Model Output: A scorecard for predicting heart disease

Starting score (constant)	350
Age (years)	
<23	-57
23 - 32	-26
33 - 41	0
42 - 48	7
49 - 57	15
58 - 64	24
65 - 71	31
>71	65
BMI (weight in kg / {height in metres}^{2})	
<19	2
19 - 26	0
27 - 29	8
30 - 32	14
>32	29
Gender	
Male	2
Female	-4
Alcohol consumption (units/week)	
0	4
1 - 12	0
13 - 24	5
25 - 48	10
Gross annual income ($)	
< $22,000	11
$22,001 - $38,000	6
$38,001 - $60,000	0
$60,001 - $94,000	-3
$94,001 - $144,000	-5
>$144,000	-6
Smoker ?	
Yes	37
No	0
Diabetic ?	
Yes	21
No	0
Cholesterol level (mg per decilitre of blood)	
Low (< 160 mg)	-2
Normal (160 - 200 mg)	0
High (201 - 240 mg)	19
Very high (>240 mg)	32
Blood pressure	
Low (below 90/60)	3
Average (between 90/60 and 140/90)	0
High (above 140/90)	36

Fig B. A score distribution (propensity) table

Group	Score range From	Score range To	Number of people	% of population	Number with heart disease after	% with heart disease after 5 yrs.
1	0	300	55,950	11.19%	40	0.07%
2	301	320	56,606	11.32%	68	0.12%
3	321	340	59,700	11.94%	129	0.22%
4	341	360	58,706	11.74%	216	0.37%
5	361	380	64,429	12.89%	403	0.63%
6	381	400	52,749	10.55%	575	1.09%
7	401	420	34,089	6.82%	600	1.76%
8	421	440	21,107	4.22%	632	2.99%
9	441	460	17,269	3.45%	878	5.09%
10	461	480	23,364	4.67%	2,020	8.65%
11	481	500	17,477	3.50%	2,553	14.61%
12	501	520	13,554	2.71%	3,366	24.84%
13	521	540	7,103	1.42%	3,463	48.76%
14	541	560	8,260	1.65%	6,587	79.74%
15	561	999	9,637	1.93%	8,469	87.88%
Total	**Total**		**500,000**		**30,000**	**6.0%**

For Group 2, the model prediction is 0.12% (=68/56,606) ➔ about a 1 in 833 (= 1/0.0012) chance of developing heart disease in the next 5 years

Fig C. Extended score distribution (propensity) table

							Descending cumulative				
Group	Score range From	To	Number of people	% of population	Number with heart disease after 5 yrs.	% with heart disease after 5 yrs.	Number of people	% of population	Number with heart disease after	% with heart disease after 5 yrs.	Purity (Lift)
1	0	300	55,950	11.19%	40	0.07%	500,000	100.00%	30,000	6.00%	6.00%
2	301	320	56,606	11.32%	68	0.12%	444,050	88.81%	29,960	6.75%	6.75%
3	321	340	59,700	11.94%	129	0.22%	387,444	77.49%	29,892	7.72%	7.72%
4	341	360	58,706	11.74%	216	0.37%	327,744	65.55%	29,763	9.08%	9.08%
5	361	380	64,429	12.89%	403	0.63%	269,038	53.81%	29,547	10.98%	10.98%
6	381	400	52,749	10.55%	575	1.09%	204,609	40.92%	29,144	14.24%	14.24%
7	401	420	34,089	6.82%	600	1.76%	151,860	30.37%	28,569	18.81%	18.81%
8	421	440	21,107	4.22%	632	2.99%	117,771	23.55%	27,969	23.75%	23.75%
9	441	460	17,269	3.45%	878	5.09%	96,664	19.33%	27,337	28.28%	28.28%
10	461	480	23,364	4.67%	2,020	8.65%	79,395	15.88%	26,459	33.33%	33.33%
11	481	500	17,477	3.50%	2,553	14.61%	56,031	11.21%	24,439	43.62%	43.62%
12	501	520	13,554	2.71%	3,366	24.84%	38,554	7.71%	21,885	56.77%	56.77%
13	521	540	7,103	1.42%	3,463	48.76%	25,000	5.00%	18,519	74.08%	74.08%
14	541	560	8,260	1.65%	6,587	79.74%	17,897	3.58%	15,056	84.12%	84.12%
15	561	999	9,637	1.93%	8,469	87.88%	9,637	1.93%	8,469	87.88%	87.88%
Total			500,000		30,000	6.0%					

Interpreting Results:

Back to the business Goals

- *Model Evaluation*: how accurate are predictions?
  - Compare Grp 15 to Grp 1:
    88%/0.07% = 1,2555
  - Compare Grp 15 to naïve rule:
    88%/6% = 15

- *Constraint:* Doctors can see 1 in 20 people (5%).
  Which 5% of patients to invite?:
  - 1.93% scores 561 or more.
    3.58% scores 541 or more.
    5.00% scores 521 or more.
    7.71% scores 501 or more. etc.

- **Decision rule**: Invite people with score > 521
  - *Impact of the rule* :
    18,519 (3,463 + 6,587 + 8,469) cases scoring 521 or more = 62% of the total 30,000.
    ➔ Selecting 5% will identify 62% of potential case, better than our goal of 50%

Fig C. Extended score distribution (propensity) table

							Descending cumulative				
Group	Score range From	Score range To	Number of people	% of population	Number with heart disease after 5 yrs.	% with heart disease after 5 yrs.	Number of people	% of population	Number with heart disease after	% with heart disease after 5 yrs.	Purity (Lift)
1	0	300	55,950	11.19%	40	0.07%	500,000	100.00%	30,000	6.00%	6.00%
2	301	320	56,606	11.32%	68	0.12%	444,050	88.81%	29,960	6.75%	6.75%
3	321	340	59,700	11.94%	129	0.22%	387,444	77.49%	29,892	7.72%	7.72%
4	341	360	58,706	11.74%	216	0.37%	327,744	65.55%	29,763	9.08%	9.08%
5	361	380	64,429	12.89%	403	0.63%	269,038	53.81%	29,547	10.98%	10.98%
6	381	400	52,749	10.55%	575	1.09%	204,609	40.92%	29,144	14.24%	14.24%
7	401	420	34,089	6.82%	600	1.76%	151,860	30.37%	28,569	18.81%	18.81%
8	421	440	21,107	4.22%	632	2.99%	117,771	23.55%	27,969	23.75%	23.75%
9	441	460	17,269	3.45%	878	5.09%	96,664	19.33%	27,337	28.28%	28.28%
10	461	480	23,364	4.67%	2,020	8.65%	79,395	15.88%	26,459	33.33%	33.33%
11	481	500	17,477	3.50%	2,553	14.61%	56,031	11.21%	24,439	43.62%	43.62%
12	501	520	13,554	2.71%	3,366	24.84%	38,554	7.71%	21,885	56.77%	56.77%
13	521	540	7,103	1.42%	3,463	48.76%	25,000	5.00%	18,519	74.08%	74.08%
14	541	560	8,260	1.65%	6,587	79.74%	17,897	3.58%	15,056	84.12%	84.12%
15	561	999	9,637	1.93%	8,469	87.88%	9,637	1.93%	8,469	87.88%	87.88%
Total			500,000		30,000	6.0%					

Interpreting Results:

Back to the business Goals

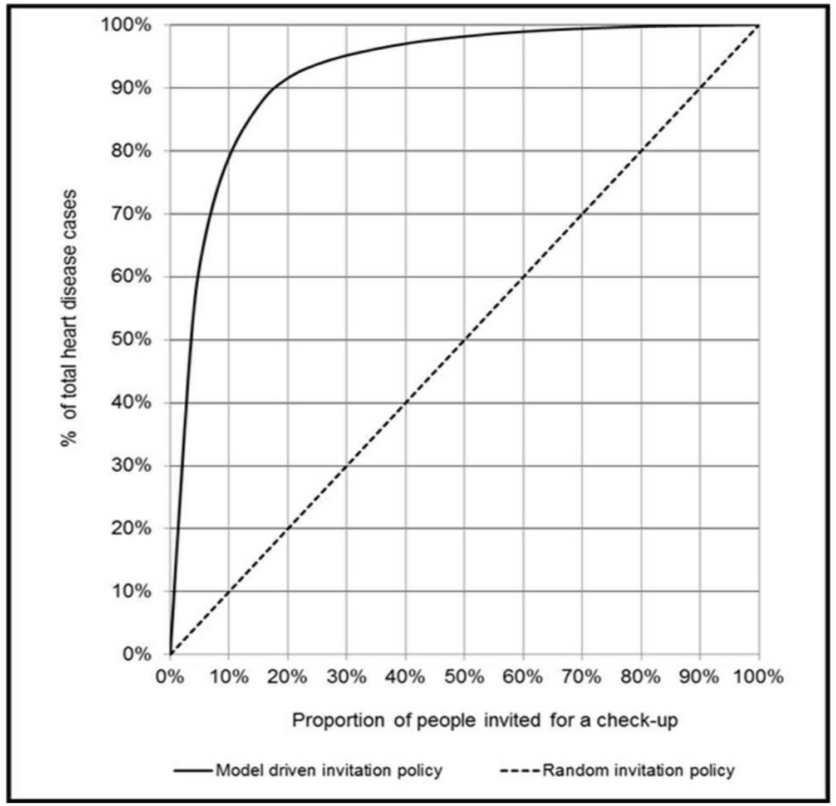
- Decision rule: Invite people with score > 521
  - *Impact of the rule* : 18,519 (3,463 + 6,587 + 8,469) cases scoring 521 or more = 62% of the total 30,000.
    ➔ Selecting 5% will identify 62% of potential case, better than our goal of 50%
  - Hit rate: There are 25,000 cases scoring 521 or more. Of these, 18,519 get heart disease. The overall hit rate is therefore 74% (100 * 18,519/25,000). 26% of people invited don't really need a check-up.
  - The rule is not perfect; gets it wrong 26% of the time, but it performs far better than the naïve rule. With only 6% of the population expected to develop heart disease in the next five years, a random invitation strategy would result in 94% (100% – 6%) of wasted check-ups!

Fig C. Extended score distribution (propensity) table

							Descending cumulative				
	Score range		Number of	% of	Number with	% with heart	Number of	% of	Number with	% with heart	Purity
Group	From	To	people	population	heart disease after 5 yrs.	disease after 5 yrs.	people	population	heart disease after	disease after 5 yrs.	(Lift)
1	0	300	55,950	11.19%	40	0.07%	500,000	100.00%	30,000	6.00%	6.00%
2	301	320	56,606	11.32%	68	0.12%	444,050	88.81%	29,960	6.75%	6.75%
3	321	340	59,700	11.94%	129	0.22%	387,444	77.49%	29,892	7.72%	7.72%
4	341	360	58,706	11.74%	216	0.37%	327,744	65.55%	29,763	9.08%	9.08%
5	361	380	64,429	12.89%	403	0.63%	269,038	53.81%	29,547	10.98%	10.98%
6	381	400	52,749	10.55%	575	1.09%	204,609	40.92%	29,144	14.24%	14.24%
7	401	420	34,089	6.82%	600	1.76%	151,860	30.37%	28,569	18.81%	18.81%
8	421	440	21,107	4.22%	632	2.99%	117,771	23.55%	27,969	23.75%	23.75%
9	441	460	17,269	3.45%	878	5.09%	96,664	19.33%	27,337	28.28%	28.28%
10	461	480	23,364	4.67%	2,020	8.65%	79,395	15.88%	26,459	33.33%	33.33%
11	481	500	17,477	3.50%	2,553	14.61%	56,031	11.21%	24,439	43.62%	43.62%
12	501	520	13,554	2.71%	3,366	24.84%	38,554	7.71%	21,885	56.77%	56.77%
13	521	540	7,103	1.42%	3,463	48.76%	25,000	5.00%	18,519	74.08%	74.08%
14	541	560	8,260	1.65%	6,587	79.74%	17,897	3.58%	15,056	84.12%	84.12%
15	561	999	9,637	1.93%	8,469	87.88%	9,637	1.93%	8,469	87.88%	87.88%
Total			500,000		30,000	6.0%					

- Explore several cut-offs, decision rules and trade-offs with the Gain's Chart (again!)

Fig D. Gains chart for the heart disease model



Class 3 Exercise

(Submit on canvas in today's Class Participation folder. Due before next class):

- 3.1 See slides 25 & 26: What is the chance that the 45 y/o woman who scored 404 will develop heart disease in the next 5 years? (show all calculations)

- 3.2 Complete and upload this week's lab practice files. Answer all included questions.

Next Week

- Quiz 1 (Covers Module 1 (Classes 2 & 3). See course outline for further details)

- Module 2 – Data Driven Decision Making (Unsupervised Learning:
  - Class 4. Associative Rules & Collaborative Filtering