ADS 500B - 02 - FA21 Final Project: Bank Marketing

Aaron Carr & Claire Phibbs

Initial Review of Data & Measures

- Data loaded into Python pandas dataframe from csv file
- Display dataframe to visually explore data (Fig. 1)
- Descriptive stats (Fig. 2)

Figure 2. Descriptive Statistics

	age	balance	day	duration	campaign	pdays	previous
count	43872.000000	45211.000000	45211.000000	45211.000000	45211.000000	45211.000000	45211.000000
mean	40.924781	1362.272058	15.806419	258.163080	2.763841	40.197828	0.580323
std	10.610835	3044.765829	8.322476	257.527812	3.098021	100.128746	2.303441
min	18.000000	-8019.000000	1.000000	0.000000	1.000000	-1.000000	0.000000
25%	33.000000	72.000000	8.000000	103.000000	1.000000	-1.000000	0.000000
50%	39.000000	448.000000	16.000000	180.000000	2.000000	-1.000000	0.000000
75%	48.000000	1428.000000	21.000000	319.000000	3.000000	-1.000000	0.000000
max	95.000000	102127.000000	31.000000	4918.000000	63.000000	871.000000	275.000000

- Review missing data and data types (next slide)

Figure 1. Dataframe

	age	job	marital	education	default	balance	housing	loan \
0	58.0	management	married	tertiary	no	2143	yes	no
1	44.0	technician	single	secondary	no	29	yes	no
2	33.0	entrepreneur	married	secondary	no	2	yes	yes
3	47.0	blue-collar	married	unknown	no	1506	yes	no
4	33.0	unknown	single	unknown	no	1	no	no
5	35.0	management	married	tertiary	no	231	yes	no
6	28.0	management	single	tertiary	no	447	yes	yes
7	42.0	entrepreneur	divorced	tertiary	yes	2	yes	no
8	58.0	retired	married	primary	no	121	yes	no
9	43.0	technician	single	secondary	no	593	yes	no

	contact	day	month	duration	campaign	pdays	previous	poutcome	deposit
0	unknown	5	may	261	1	-1	0	unknown	no
1	unknown	5	may	151	1	-1	0	unknown	no
2	unknown	5	may	76	1	-1	0	unknown	no
3	unknown	5	may	92	1	-1	0	unknown	no
4	NaN	5	may	198	1	-1	0	unknown	no
5	unknown	5	may	139	1	-1	0	unknown	no
6	unknown	5	may	217	1	-1	0	unknown	no
7	unknown	5	may	380	1	-1	0	unknown	no
8	unknown	5	may	50	1	-1	0	unknown	no
9	unknown	5	may	55	1	-1	0	unknown	no

Bank Marketing Dataset Characteristics

Variable Types

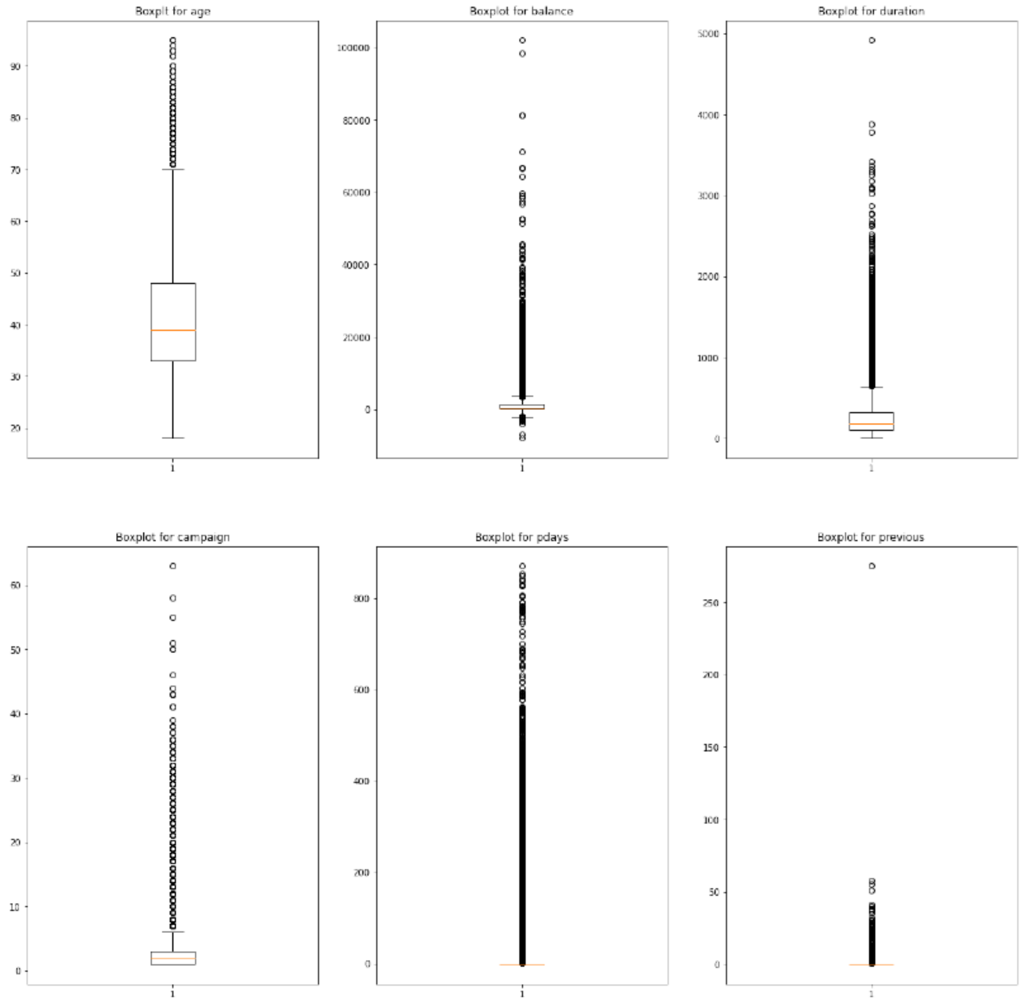
```
age          float64
job           object
marital       object
education     object
default       object
balance        int64
housing       object
loan          object
contact       object
day            int64
month         object
duration       int64
campaign       int64
pdays          int64
previous       int64
poutcome      object
deposit       object
dtype: object
```

Null Value Count by Variable

```
age          1339
job             0
marital         0
education       0
default      1306
balance         0
housing         0
loan            0
contact      1383
day             0
month           0
duration        0
campaign        0
pdays           0
previous        0
poutcome        0
deposit         0
dtype: int64
```

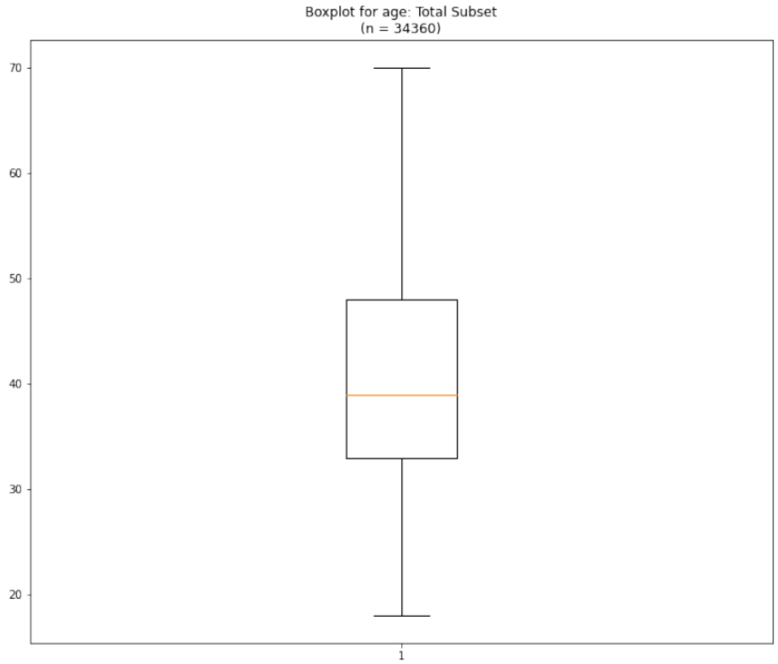
Initial Visualizations

- Review Boxplots
  - Plot all numerical vars together
  - Check variance, IQR, and outliers



Cleaning & Transforming Data

- Fill in missing data using appropriate methods
  - E.g., null 'age' values filled in using mean substitution by grouped aggregation on marital status & education level
- Eliminate columns not needed for analysis (e.g., 'contact')
- Create new features to simplify analysis
  - New feature 'previous_contact' (yes/no) based on 'pdays' = -1
- Discretize categorical variables
  - Create dummy variables, otherwise
- Transform ambiguous values
  - E.g., change "unknown" for education based on mode value of group aggregates
- Final boxplot & data reviews



Code Example: Define Fx & Run Transx

Define function: Fill in missing numerical data based on group by {-}

```python
def gb_agg_sub(df, t_var=None, gb_vars=[], agg_meth='mean'):
    '''current aggregate methods = sum, mean'''
    if agg_meth == 'sum':
        df_gpb01 = df.groupby(gb_vars).sum() # create a multi-indexed dataframe
    else:
        df_gpb01 = df.groupby(gb_vars).mean() # create a multi-indexed dataframe
    print(df_gpb01)
    df = pd.merge(df, df_gpb01, how='left', on=gb_vars, suffixes=(None, '_y'))
    df[t_var] = df[t_var].fillna(value=df[t_var + '_y'])
    return df
```

```python
# Run function to fill in missing age values based on group by
bank_df01 = gb_agg_sub(bank_df01, 'age',  ['marital', 'education'])

# Reset col names
bank_df01 = bank_df01[bank_df01_cols_lst01]

# Remove records where balance is less than zero
bank_df01 = bank_df01.loc[(bank_df01['balance'] >= 0), :]

# Create new feature
bank_df01['previous_contact'] = 0
bank_df01.loc[(bank_df01['pdays'] != -1), 'previous_contact'] = 1

# Save df len for generating % loss later
bank_df01_len02 = len(bank_df01)

# Remove col not used for analysis
bank_df01 = bank_df01.drop(['contact'], axis=1)
bank_df01 = bank_df01.drop(['pdays'], axis=1)
bank_df01 = bank_df01.drop(['previous'], axis=1)

# Fill in `default` w/ "unknown" to transform it below
bank_df01['default'] = bank_df01['default'].fillna(unk_str)

print(bank_df01.head())
```

Post-processing Review

After cleaning and trimming the data perform additional reviews

Figure 3. Descriptive Statistics

	age	balance	day	duration	campaign	previous_contact
count	34360.000000	34360.000000	34360.000000	34360.000000	34360.000000	34360.000000
mean	40.395460	775.747875	15.408615	261.549447	2.130850	0.191473
std	9.908841	879.924490	8.265127	256.529050	1.315766	0.393466
min	18.000000	0.000000	1.000000	0.000000	1.000000	0.000000
25%	33.000000	119.000000	8.000000	107.000000	1.000000	0.000000
50%	39.000000	439.000000	15.000000	184.000000	2.000000	0.000000
75%	48.000000	1128.000000	21.000000	322.000000	3.000000	0.000000
max	70.000000	3770.000000	31.000000	3881.000000	6.000000	1.000000

Figure 4. Correlation Matrix

	age	balance	day	duration	campaign	previous_contact
age	1.000000	0.086771	-0.008327	-0.016820	0.039928	-0.018423
balance	0.086771	1.000000	0.021369	0.041036	-0.023945	0.054869
day	-0.008327	0.021369	1.000000	-0.017647	0.101958	-0.069389
duration	-0.016820	0.041036	-0.017647	1.000000	-0.027469	-0.004868
campaign	0.039928	-0.023945	0.101958	-0.027469	1.000000	-0.094269
previous_contact	-0.018423	0.054869	-0.069389	-0.004868	-0.094269	1.000000

Formulas for Logistic Regression Models

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Logit Function:

$$P(Y = 1|X) = \frac{\mathrm{e}^{(\beta_o + \beta_1 x)}}{\mathrm{e}^{(\beta_o + \beta_1 x)} + 1}$$

ROC: True Positive Rate ~ False Positive Rate

Model 1: Deposit~previous_contact

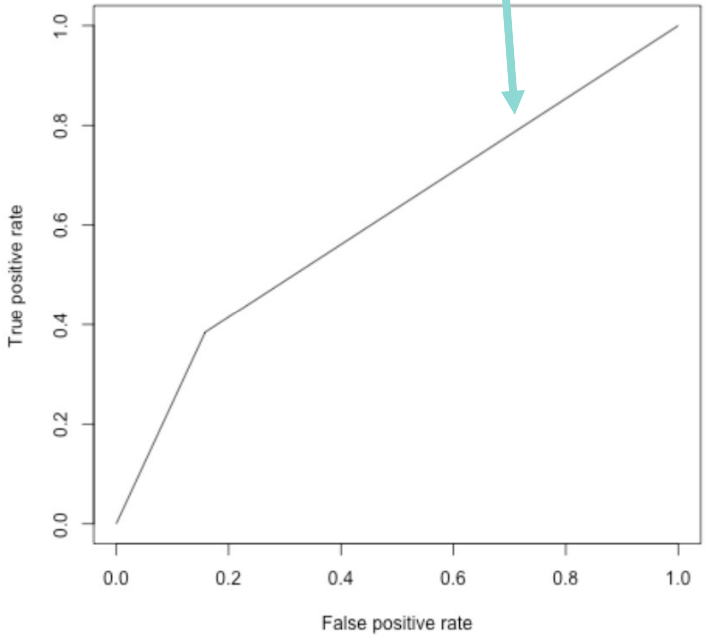
```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.7189  -0.4447  -0.4447  -0.4447   2.1740

Coefficients:
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)      -2.26421    0.02459  -92.06   <2e-16 ***
previous_contact  1.04288    0.04271   24.42   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

*Models Predictive Ability→ weak due to only (1) predictor

AUC=0.6131

Accuracy=0.8786



Model 4: Deposit~age+72 other predictor variables (Full Model)

List of Perfectly Linearly Dependent Variables:

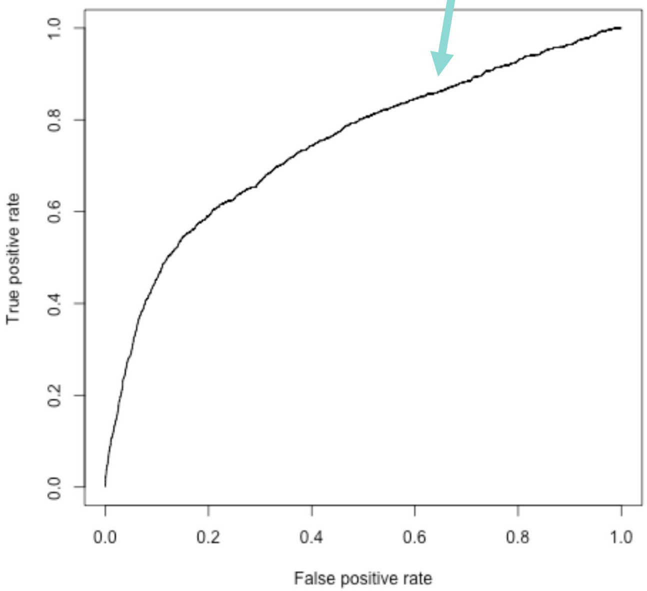
- Job_housemaid
- Marital_divorced
- Education_unknown
- Default_no
- Housing_no
- Loan_yes
- Day_31
- Month_sep

These variables did not output any coefficients due to their “perfect” linear dependence. They have been removed in the next model.

*Models Predictive Ability→ moderate due to high number of predictors but not accounting for linear dependence

AUC=0.7458

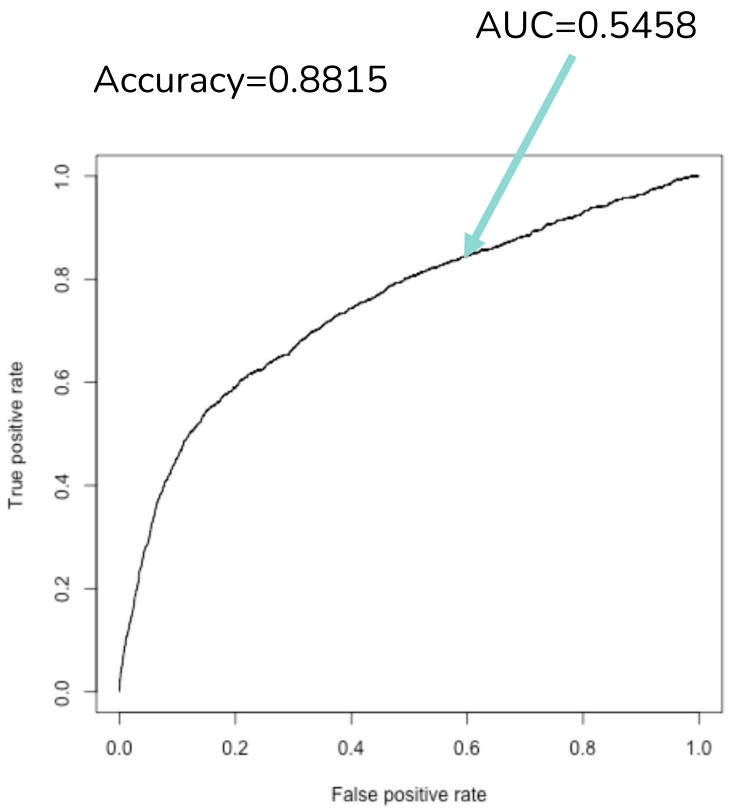
Accuracy=0.8815



Model 5: Deposit~age+64 other predictor variables (Trimmed Model)

VIFs calculated to check linear dependence of predictors → many predictors with VIF>2.5

*Models Predictive Ability→ relatively weak due to the larger number of predictors used but some of them still exhibiting high degree of linear dependence.



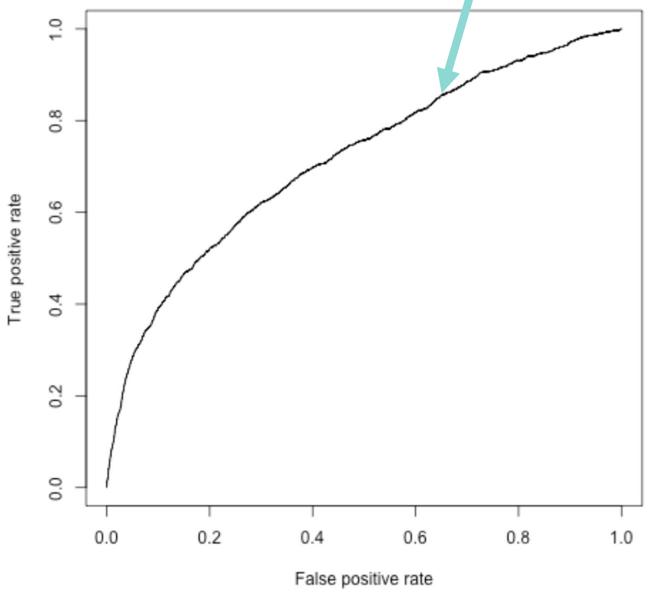
Model 7: Deposit~age+balance+campaign+previous_contact+job_entrepreneur+housing_yes+loan_no+month_dec+month_mar

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.8319  -0.5447  -0.3914  -0.3125   2.9007

Coefficients:
                   Estimate Std. Error z value Pr(>|z|)
(Intercept)      -1.8209708  0.1147669 -15.867  < 2e-16 ***
age              -0.0095439  0.0020300  -4.701 2.58e-06 ***
balance           0.0001948  0.0000216   9.019  < 2e-16 ***
campaign         -0.1267321  0.0172572  -7.344 2.08e-13 ***
previous_contact  1.0644249  0.0452100  23.544  < 2e-16 ***
job_entrepreneur -0.2937912  0.1332428  -2.205   0.0275 *
housing_yes      -0.9977238  0.0433240 -23.029  < 2e-16 ***
loan_no           0.5373725  0.0701023   7.666 1.78e-14 ***
month_dec         0.9277300  0.2048106   4.530 5.91e-06 ***
month_mar         1.5968905  0.1361512  11.729  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Accuracy=0.8805

AUC=0.7150



*Models Predictive Ability→ moderate due to predictors used & significance of predictors

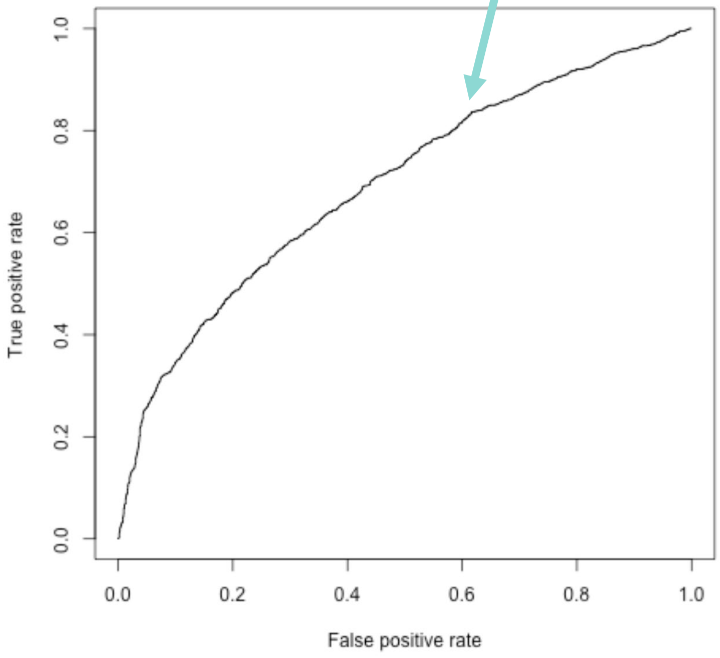
Model 7b: Final Model on Subset Where "yes" ~40% of the Sample

AUC=0.6935

Accuracy=0.6814

```
Deviance Residuals:
    Min       1Q    Median       3Q      Max
-2.6641  -0.9441  -0.6709   1.1395   2.2519

Coefficients:
                   Estimate Std. Error z value Pr(>|z|)
(Intercept)      -9.025e-02  1.385e-01  -0.652    0.515
age              -1.166e-02  2.502e-03  -4.660 3.17e-06 ***
balance           1.914e-04  2.793e-05   6.856 7.10e-12 ***
campaign         -1.522e-01  2.097e-02  -7.258 3.93e-13 ***
previous_contact  1.092e+00  5.944e-02  18.372  < 2e-16 ***
job_entrepreneur -1.276e-01  1.655e-01  -0.771    0.441
housing_yes      -9.242e-01  5.294e-02 -17.458  < 2e-16 ***
loan_no           4.666e-01  8.240e-02   5.662 1.49e-08 ***
month_dec         1.255e+00  3.002e-01   4.181 2.90e-05 ***
month_mar         2.292e+00  2.650e-01   8.649  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```



*Models Predictive Ability→ Similar to model 7, but the AUC and accuracy actually decreased

Concluding Thoughts

- Based on full EDA and logistic regression, trimmed Model 7 predicts deposit status the best compared to the other models provided.
- Study strengths included a large dataset and integration of Python & R
- Limitations due to data make up, e.g., ratios of values & large number of categorical variables

References

- Coban, H. (2019). Here's how I used Python to build a regression model using an e-commerce dataset. https://searchengineland.com/heres-how-i-used-python-to-build-a-regression-model-using-an-e-commerce-dataset-326493
- Devore, J. L. (2016). *Probability and statistics*. (9th ed.). Cengage Learning.
- Mukala, M. M. (2012). A guide to appropriate use of correlation coefficient in medical research. Malawi Medical Journal, *24*(3), 69-71.
- Real Python. (n.d.). Linear regression in python. https://realpython.com/linear-regression-in-python/
- Real Python. (n.d.). Logistic regression in python. https://realpython.com/logistic-regression-python/
- Shah, C. (2020). *A hands-on introduction to data science*. Cambridge University Press.
- Stack Overflow. (n.d.). GroupBy pandas dataFrame and select most common value. https://stackoverflow.com/questions/15222754/groupby-pandas-dataframe-and-select-most-common-value
- UCLA: Statistical Consulting Group. Introduction to SAS. https://stats.idre.ucla.edu/sas/modules/sas-learning-moduleintroduction-to-the-features-of-sas/ (accessed December 12, 2021).

ADS 500B - 02 - FA21 Final Project: Bank Marketing

Aaron Carr & Claire Phibbs