UC San Diego

# Intuit Quickbooks

Group 14 - Aditya Kumawat
Bindu Priyanka Achalla
Keerthana Raviprasad
Yves Kalin

# Introduction:

Objective: Predict small business responses to a QuickBooks upgrade campaign, using data from 75,000 businesses out of an initial 801,821 to enhance marketing efficiency.

Approach: Employ data manipulation and logistic regression for predictive modeling, aiming to pinpoint businesses likely to upgrade, thereby optimizing Intuit's marketing strategy and profitability.

# Preprocessing:

- Feature Scaling and Normalization: Ensures each feature contributes equally by scaling data, improving model accuracy and performance. Handling Missing Values and Encoding
- Categorical Variables: Fills in missing data and converts categories to numerical formats, making the dataset compatible with logistic regression.
- Feature Engineering and Outlier Removal: Enhances model capability by creating new features and removing data points that significantly deviate from others, preventing model distortion.
- Multicollinearity Detection: Addresses high correlations between predictors to ensure independent variable assumptions are met, enhancing model reliability.

# Preprocessing:

- Categorical values :

```
intuit75k['zip_bins_cat'] =  intuit75k['zip_bins'].astype('category')
intuit75k['upgraded'] =  intuit75k['upgraded'].astype('category')
intuit75k['version1'] =  intuit75k['version1'].astype('category')
intuit75k['owntaxprod'] =  intuit75k['owntaxprod'].astype('category')
intuit75k['bizflag'] =  intuit75k['bizflag'].astype('category')
```

- Handling NA values:

```
na_values = intuit75k.isna()
print(na_values.sum())
```

# Preprocessing:

- Scale_df (pyrsm)

```
def scale_df(
    df, wt=None, sf=2, excl=None, train=None, ddof=0, stats=False, means=None, stds=None
):
    df = df.copy()
    isNum = [
        col
        for col in df.columns
        if pd.api.types.is_numeric_dtype(df[col].dtype)
        and not (df[col].nunique() == 2 and df[col].min() == 0 and df[col].max() == 1)
    ]

    if excl is not None:
        isNum = setdiff(isNum, excl)
    dfs = df[isNum]
    if train is None:
        train = np.array([True] * df.shape[0])
    if wt is None:
        if means is None:
            means = dfs[train].mean().values
        else:
            means = np.array([means[c] for c in isNum])
        if stds is None:
            stds = sf * dfs[train].std(ddof=ddof).values
        else:
            stds = np.array([stds[c] for c in isNum])
        df.loc[:, isNum] = (dfs - means) / stds

    else:
        means = weighted_mean(dfs[train], wt[train])
        stds = sf * weighted_sd(dfs[train], wt[train], ddof=ddof)
        wt = np.array(wt)
        df.loc[:, isNum] = (dfs - means) / stds

    if stats:
        means = {c: means[i] for i, c in enumerate(isNum)}
        stds = {c: stds[i] for i, c in enumerate(isNum)}
        return df, means, stds
    else:
        return df
```

# Approach:

- Logistic Regression - pyrsm
- Neural Network  (MLP classifier)  - sklearn
- Grid Search Method - sklearn

# Analysis (Logistic Basic Model):

Analysis revealed that among all the predictors included, only sex, bizflag and sincepurch demonstrated statistical insignificance. All other variables are significant.
The first logistic regression with all the significant predictors, was statistically significant with p value of < 0.001.
Pseudo R-Sqaured: 0.114, Train AUC 0.755 and Test AUC 0.7546.

.

```
lr1 = rsm.model.logistic(
    data={"intuit":intuit75k[intuit75k['training'] == 1]}, rvar="res1", lev='Yes', evar=['zip_bins_cat','dollars',
                                                                                        'numords','last','version1','owntaxprod','upgraded'],
)
lr1.summary(vif=True)
```

# Analysis (Neural Network):

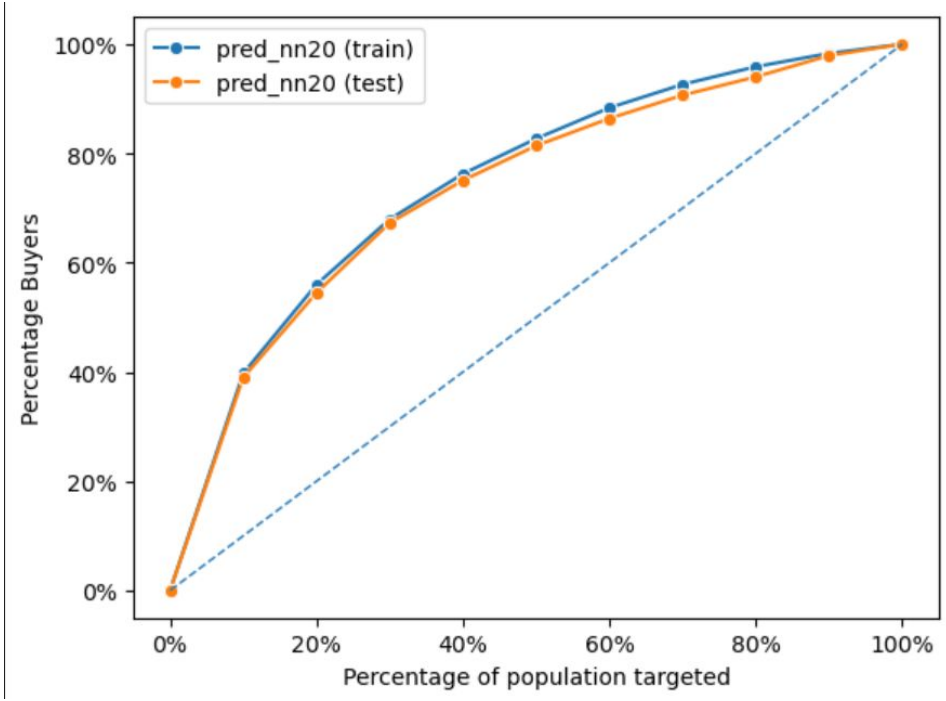
UC San Diego

We employed a grid search to identify the neural network configuration that yielded the highest AUC (Area Under the ROC Curve). Among top 5 NN models, we compared results of different models using Test AUC, predicted profit on test data. We decided to use the following model with 5 nodes and alpha 10.

Train AUC: 0.767, Test AUC: 0.7556

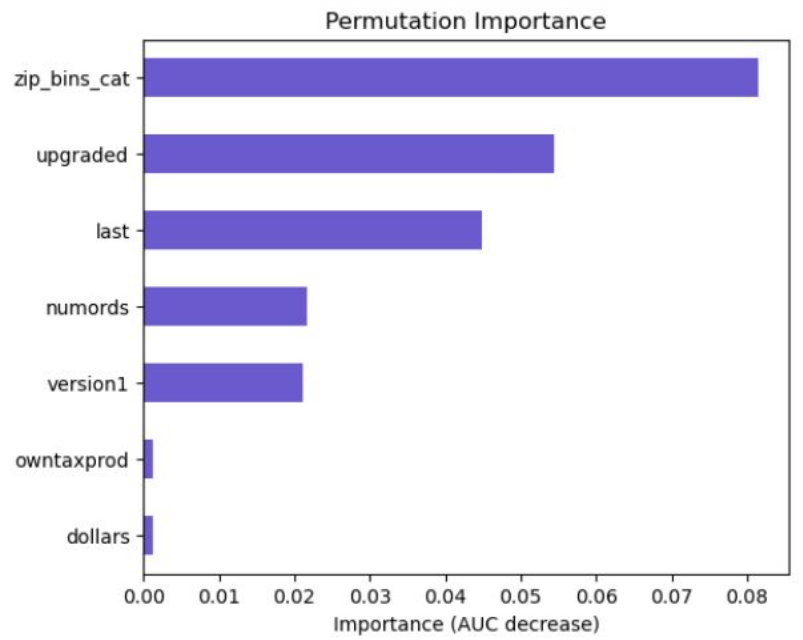
```
Multi-layer Perceptron (NN)
Data                : data
Response variable   : res1_yes
Level               : 1
Explanatory variables: zip_bins_cat, numords, dollars, last, version1, owntaxprod, upgraded, sex, bizflag, sincepurch
Model type          : classification
Hidden_layer_sizes  : (5,)
Activation function : tanh
Solver              : lbfgs
Alpha               : 10
Batch size          : auto
Learning rate       : 0.001
Maximum itterations : 10000
random_state        : 1234
AUC                 : 0.767
```

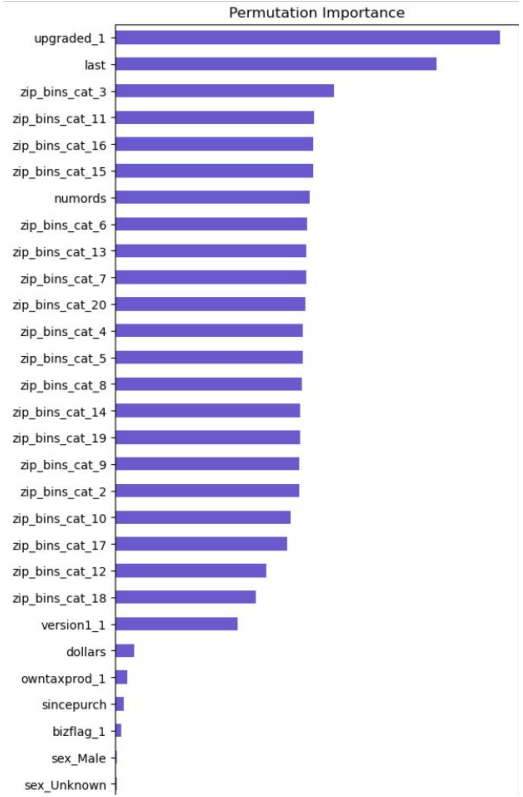
# Analysis (Neural Network):




The gains chart show that there was no overfit in the Neural Network Model.

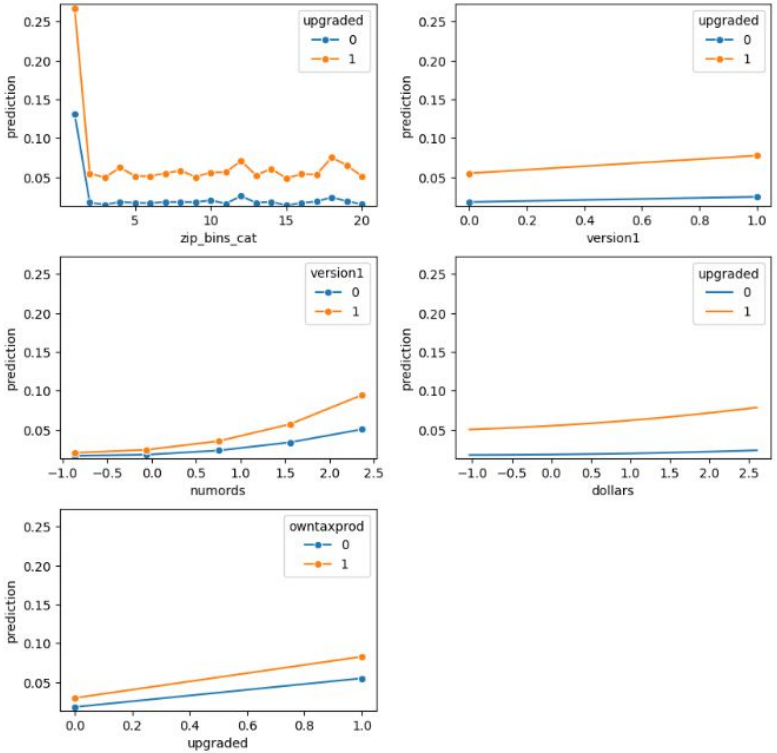
# Permutation Plots (Neural Network Vs Basic Logistic):




Upgraded and last are on the top for NN, whereas zipcodes are on the top for Basic Logistic Model.

# Interactions




We reviewed various interaction charts to explore potential interactions between variables. Our observations indicated that several charts displayed non-parallel line.

To intercept one of these: The first plot clearly indicate that the predicted probability of responding to the first mail changes more significantly in some zip codes (against other zipcodes)  when compared to the impact of a business's upgrade status.

# Enhancing Logistic Model

UC San Diego

These identified interaction variables were incorporated into different logistic regression models for further analysis. All the different enhanced logistic regression models were evaluated on the basis of test AUC and expected profit on Test set.
We were able to enhance the Test AUC and expected profit compared to the Basic Logistic Model. Below are two such models where Test AUC improved

```
lr2 = rsm.model.logistic(
    data={"intuit":intuit75k[intuit75k['training'] == 1]}, rvar="res1", lev='Yes', evar=['zip_bins_cat','numords',
                                                                                         'dollars','last','version1','owntaxprod','upgraded'],
    ivar=["zip_bins_cat:upgraded", "version1:upgraded"]
)
lr2.summary(vif=True)
## Question answers
```

Train AUC: 0.758, Test AUC: 0.7554, Pseudo Rsquared: 0.116

```
lr2 = rsm.model.logistic(
    data={"intuit":intuit75k[intuit75k['training'] == 1]}, rvar="res1", lev='Yes', evar=['zip_bins_cat','numords',
                                                                                         'dollars','last','version1','owntaxprod','upgraded'],
    ivar=["zip_bins_cat:upgraded", "version1:upgraded, numords:version1"]
)
lr2.summary(vif=True)
## Question answers
```

Train AUC: 0.758, Test AUC: 0.7561, Pseudo Rsquared: 0.121

# Comparing different models:

| Model | Profit | AUC | No.of People responded |
|---|---|---|---|
| lr1 | 16132.36 | 0.070 | 5728 |
| nn20 | 17267.84 | 0.070 | 6134 |
| lr2 | 16644.21 | 0.068 | 6179 |

# Choosing the best model:

**Expected Profit Calculation for Test Set**

- Targeting individuals who didn't respond to the first mailing wave.
- Only those with a predicted response probability (adjusted by 50%) above the breakeven point will be mailed.
- The anticipated response rate for this campaign is based on the first mail wave's observed response rate.

**Model Comparison and Selection**

- The NN (5) model has the highest AUC score.
- The NN(5)  model is preferred due to its higher expected profit.

**Optimal Neural Network Configuration**

- The nnr model with 5 nodes is identified as the most balanced.
- It exhibits less overfitting compared to other configurations.

# Comparison with different strategies

| Model | Test AUC | Test Profit (Mail 1) | Puesdo - R | Test Profit Expected (Mail 2) |
|---|---|---|---|---|
| Optimized Logistic | 0.7546 | 38555 | 0.114 | 14133.71 |
| NN5(10) (best in grid search) | 0.7556 | 38214.72 | NA | 17200.72 |
| NN4(0.00001) | 0.7532 | 38466.96 | NA | 18074.554 |
| NN4(10) | 0.7544 | 38213 | NA | 17400.74 |
| "zip_bins_cat:upgraded", "version1:upgraded" (with 10 variables) Enhanced Logistic | 0.7551 | 37915.14 | 0.116 | 16687.41 |
| "zip_bins_cat:upgraded", "version1:upgraded" (with 7 variables) Enhanced Logistic | 0.7554 | 38035.14 | 0.116 | 16767.3 |
| "zip_bins_cat:upgraded", "version1:upgraded", "numords:version1" (with 7 variables) Enhanced Logistic | 0.7561 | 38463.63 | 0.121 | 16132.362 |
| zip_bins_cat:upgraded", "version1:upgraded", "numords:version1", "dollars:upgraded" (7 Variables) Enhanced Logisti | 0.7562 | 38481.96 | 0.121 | 16200.5 |
| "zip_bins_cat:upgraded", "version1:upgraded", "numords:version1", "dollars:upgraded", "upgraded:owntaxprod" (7 V | 0.7562 | 38306.82 | 0.121 | 16237.71 |

# Scaling up on 800K:

Scaled numbers for 801,821 users, subtracting 801,821 to determine actual qualified users for mail 2. Similarly, identified total qualified IDs in the test set. The ratio of the best model represents the proportion of mail 2 recipients to total qualified users in the test set. This ratio is used to calculate the number of users to receive mail based on the model among total qualified IDs. Total_responses_best_model multiplied by the response rate for mail 1.

The profit for scaling up on 800K is around 616027.07

# Calculating ID's to send mails in wave2:

To identify customers for the second wave mail from the 22,500, we select IDs with predicted probabilities greater than the breakeven point. This process yields approximately 6,134 IDs out of the 22,500, representing the targeted customers for wave-2.

| | id | mailto_wave2 |
|---|---|---|
| 2 | 3 | True |
| 19 | 20 | True |
| 46 | 47 | True |
| 54 | 55 | True |
| 71 | 72 | True |
| ... | ... | ... |
| 74964 | 74965 | True |
| 74976 | 74977 | True |
| 74979 | 74980 | True |
| 74987 | 74988 | True |
| 74999 | 75000 | True |

6134 rows × 2 columns

# Conclusion:

Model chosen: NN20 model was chosen for our final approach, effectively identifying 6,134 potential responders for the second mailing wave, factoring in the expected 50% response rate reduction from the first wave. This targeted strategy optimizes our campaign's reach and efficiency, promising to enhance profitability by concentrating on the most promising leads for the QuickBooks software upgrade.

UC San Diego

# Thank you!