# Project Introduction

This project here aims to use unsupervised machine learning techniques to segment customers and then supervised learning to predict which individuals are most likely to respond to marketing campaigns.

The project here has come about for marketing needs for Arvato's marketing campaigns. Our aims  are to use unsupervised machine learning techniques to segment customers for proposed marketing and then supervised learning to predict which individuals are most likely to respond to marketing campaigns.

## Description of Input Data

The project utilises four main datasets provided by Arvato Financial Solutions:

1. General population demographics
2. Customer demographics
3. Mailout train dataset
4. Mailout test dataset

These datasets contain demographic information on individuals, with a couple hundreds of features related to age, gender, location, housing, financial status, consumption habits, and so many more.

## Strategy for Solving the Problem

1. Extensive data preprocessing and cleaning
2. Unsupervised learning (PCA and k-means clustering) for customer segmentation
3. Supervised learning to predict customer conversion
4. Model evaluation and optimization

## Discussion of Expected Solution

In all, we expect the following:

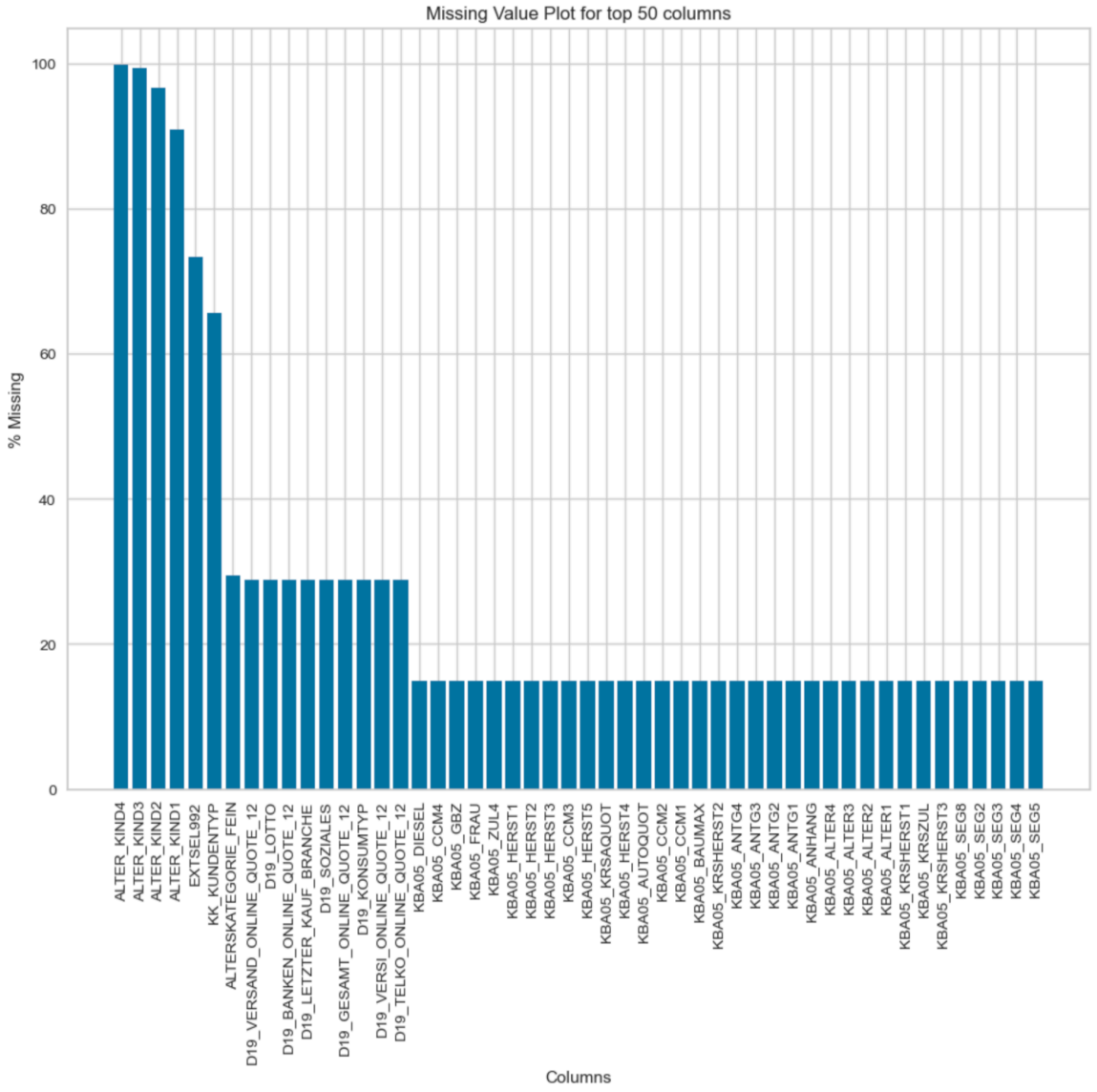
1. Cleaned and preprocessed datasets
2. PCA components capturing key demographic factors
3. Customer segments identified through clustering
4. Predictive model to identify likely customer conversions
5. Evaluation metrics and performance analysis

# Unsupervised Learning

## EDA

Exploratory data analysis revealed:

- High dimensionality with 300+ features
- Significant missing data and encoding inconsistencies
- Complex relationships between demographic variables



# Data Preprocessing

- Feature Engineering on selected columns
- Removing low variance and highly correlated features
- Addressing data inconsistencies between datasets
- Scaling all features
- Handling missing values through imputation , and in some cases, removal

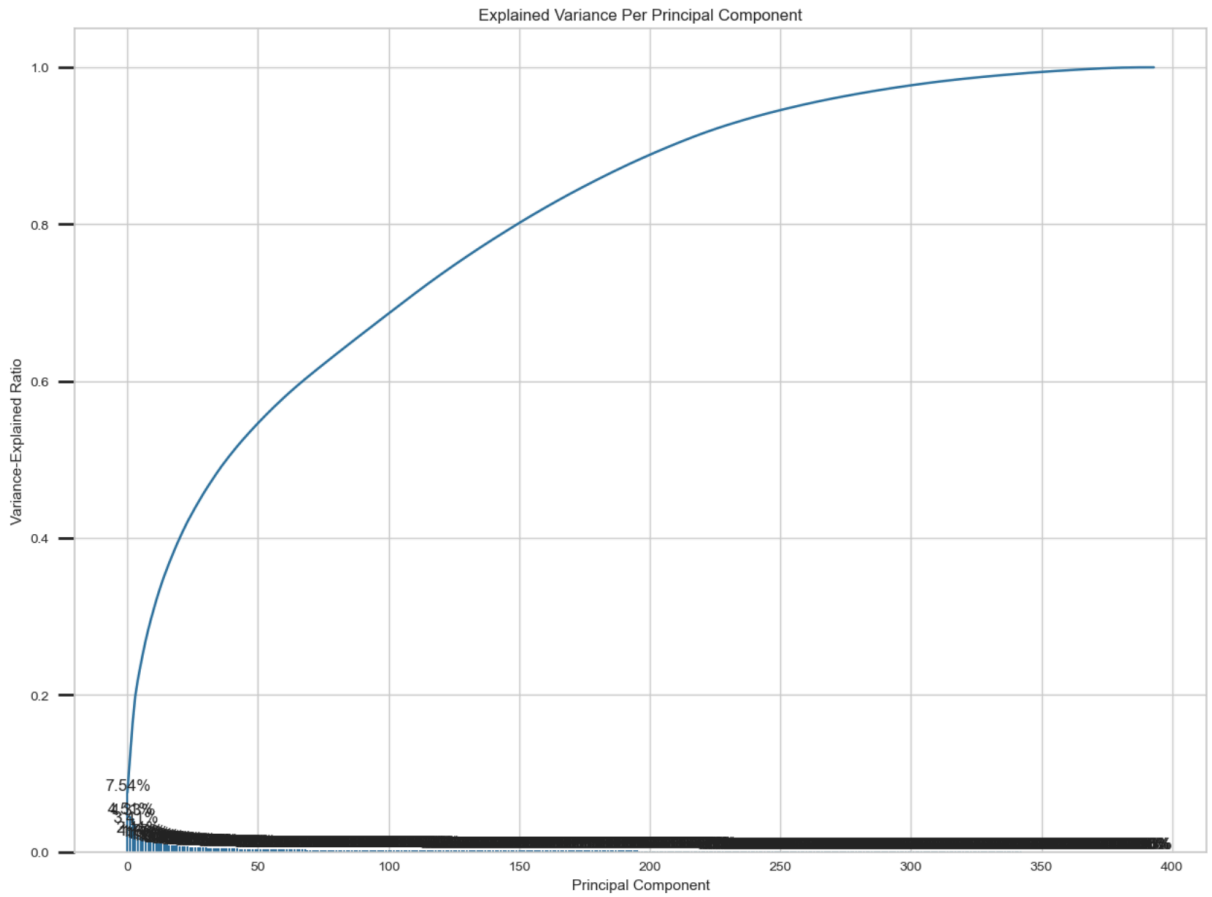
# Modelling

Key unsupervised learning steps:

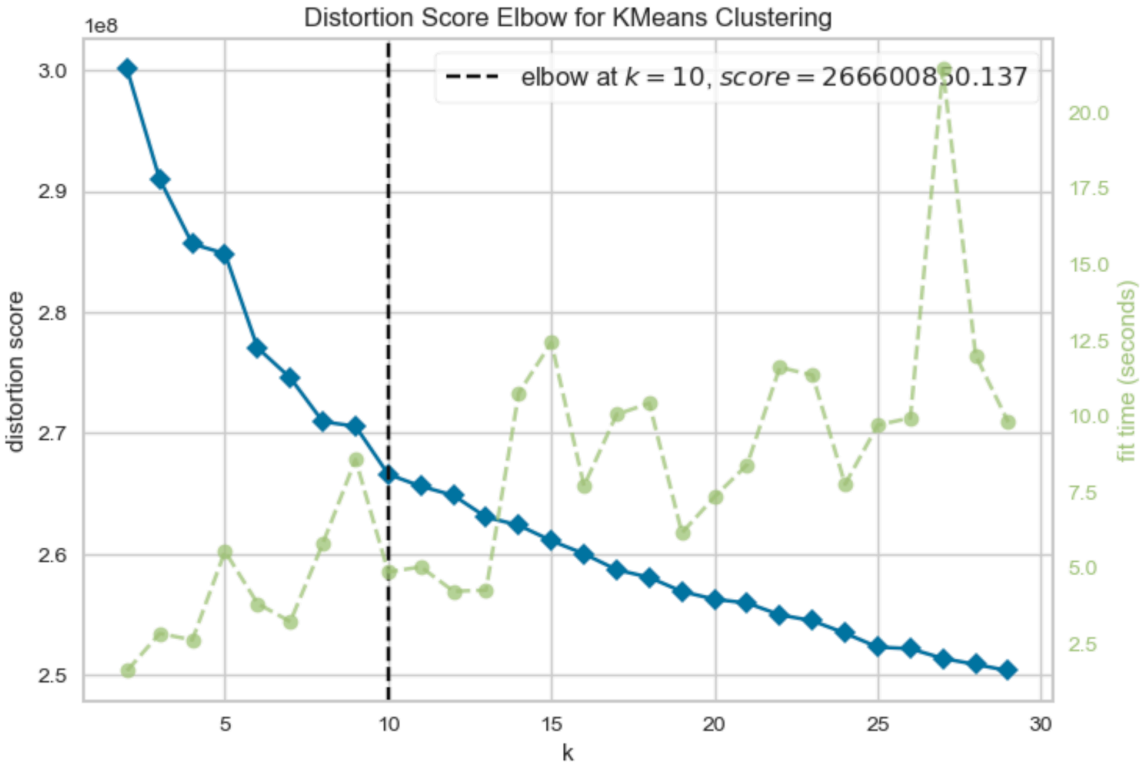
1. PCA for dimensionality reduction
2. K-means clustering for customer segmentation

# Results

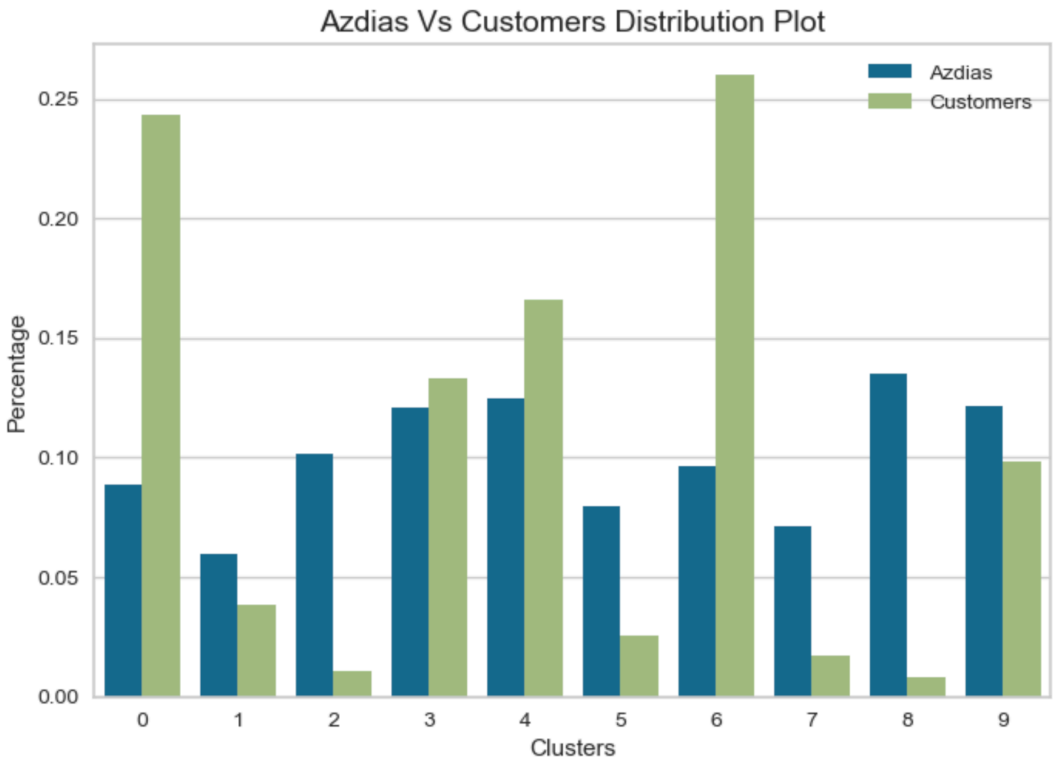
- Really, using PCA, we could see how 95% of the dataset's total variability was explained by only 215 of its components. This was great in filtering out noise.



- Moving on, using our PCA transformed dataframe, we were able to use KElbow's Visualizer to try estimate what amount of clusters to feed into our model.




- There on out, moving to the slightly anticlimactic bits, we fit our KMeans Model onto our PCA transformed dataframe, using the same number of clusters as gotten earlier.
- We then moved on to apply ALL preprocessing steps to allow a comparison between the General Data and the Customer Data.




- From the above, it was clear what cluster subsets were over, and under - represented. This is from considering the Customers dataset with respect to the General(AZDIAS).
- A further inversion of these clusters gave away what values they corresponded with in our Dataset's range of values and their meanings.

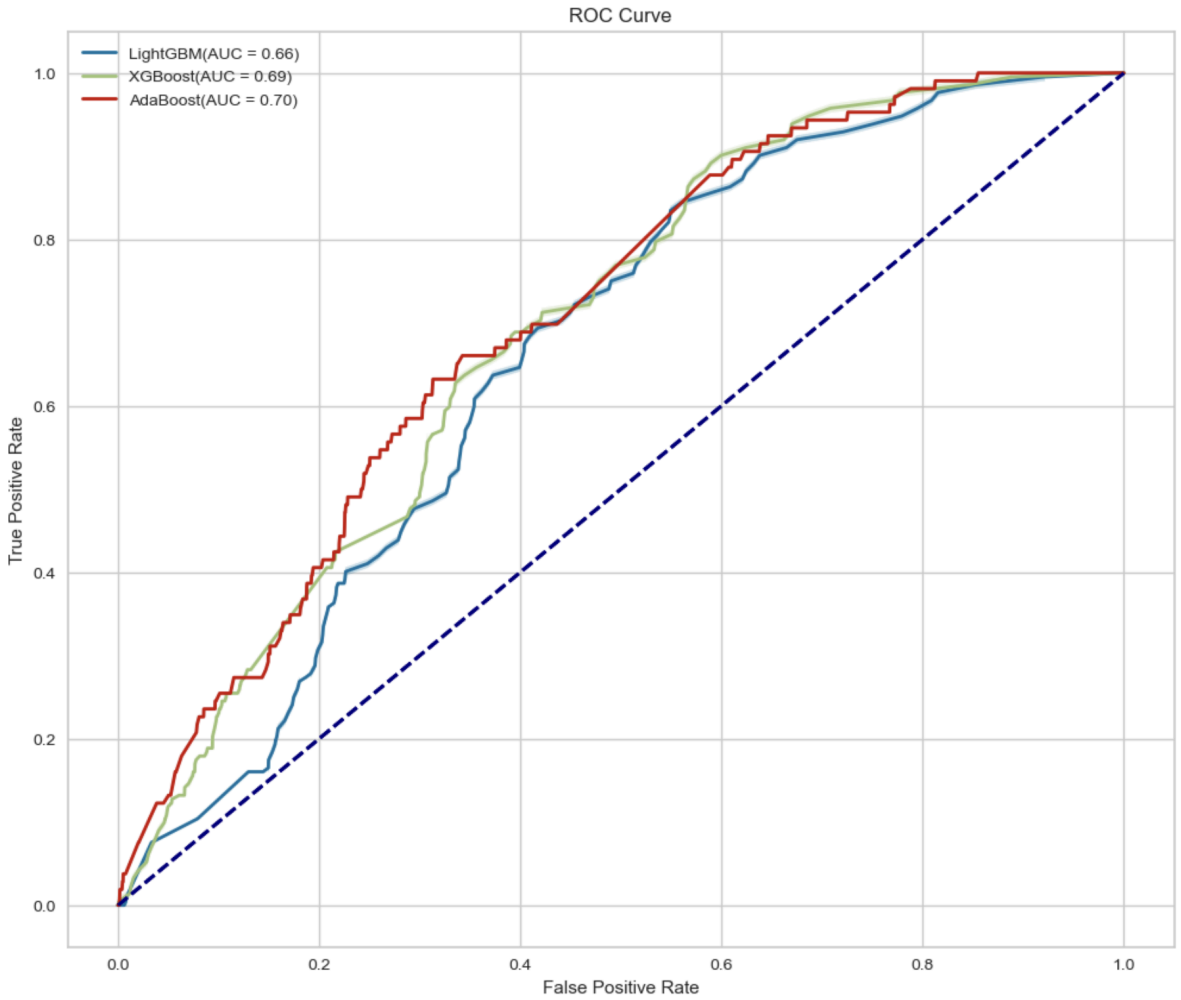
# Supervised Learning

## Modelling

Modelling after applying the same cleaning and preprocessing step to both Mailing Datasets here included first, running three boosting algorithms known to have very good industrial recognition due to stability.

- Light Gradient Boosting Machine (LGBM)
- eXtreme Gradient Boost (XGBoost)
- Adaptive Boosting (AdaBoost)

These 3 were graded based on our `Auc-Roc` Metric as required for our stakeholders here. Below is an attachment of what this comparison looked like.



## Hyperparameter Tuning

We went on to perform gradient boosting on our two best models of AdaBoost and next, XGBoost.

GridSearch with cross-validation was used for XGBoost. Key hyperparameters tuned included learning rate, max depth, scale_pos_weight and number of estimators.

While AdaBoost gave us a little less options to work with, only simply its learning rate and number of estimators.

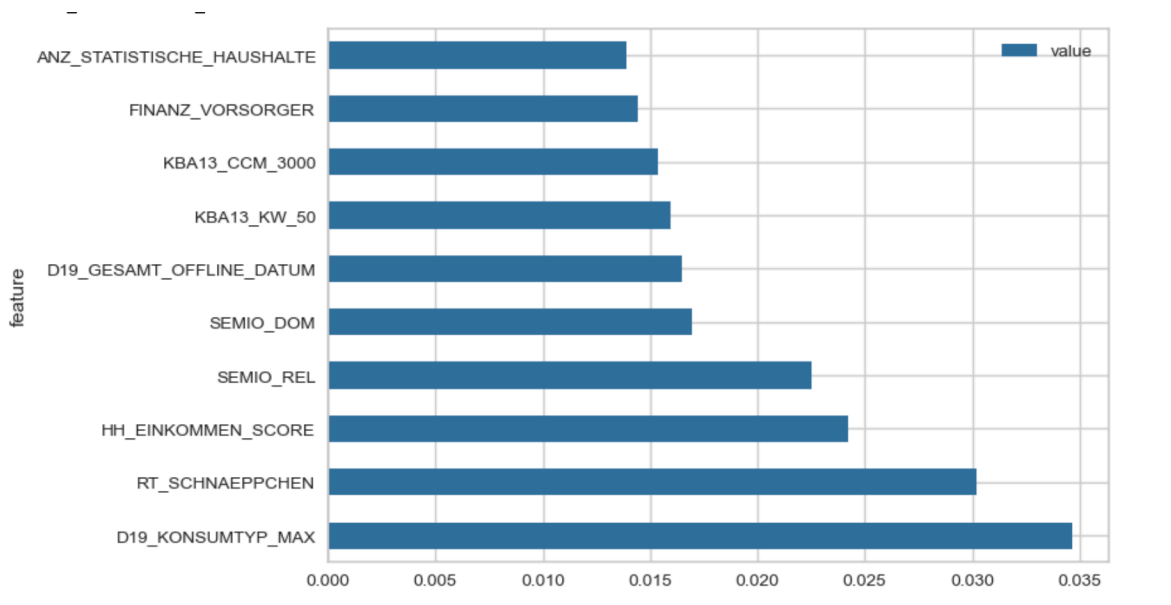
## Metric Used

AUC-ROC was the primary chosen submission metric really. Using the report here, it is known to be insensitive to class imbalance, good for us.

## Results

XGBoost classifier being tuned increased its AUC score from 0.57 to about 0.64 on the test set while AdaBoost massively dropped off to a 0.51.

We could then use this result to try to estimate what exact factors predict if or not a prospect would really key into being a customer.



## Conclusion

The project successfully identified key customer segments and developed a predictive model for customer conversion. The insights gained can help the company better target their marketing efforts. The XGBoost model provides a valuable tool for identifying high-potential customers.

## Improvements

Potential areas for improvement include:

- More advanced Feature Engineering, probably even decode as many columns as possible.
- Ensemble methods combining multiple models
- Incorporation of additional data sources
- Possibly use sampling methods such as SMOTE for class balancing

## References


1. Scikit-learn developers. (2023). Clustering. scikit-learn documentation. https://scikit-learn.org/stable/modules/clustering.html
2. Implementing Customer Segmentation using ML https://neptune.ai/blog/f1-score-accuracy-roc-auc-pr-auc
3. The XGBoost Classifier https://dmlc.cs.washington.edu/xgboost.html
4. Interpretable machine learning-based approach for customer segmentation for new product development from online product reviews https://esol.ise.illinois.edu/static2/pdf/IJIM2023.pdf


# Acknowledgments


Thanks to Arvato for providing the datasets and Udacity for the project structure and guidance.