# SANTANDER

# PRODUCT RECOMMENDATION

Chenxi (Celia) Huang

The Santander Data Group

# TODAY’S AGENDA

- **Project Description**
- **Data Processing**
- **Feature Engineering**
- **Model Selection**
- **Future Considerations**

# PROJECT DESCRIPTION

- **Project Nature:**

  Product Recommendation

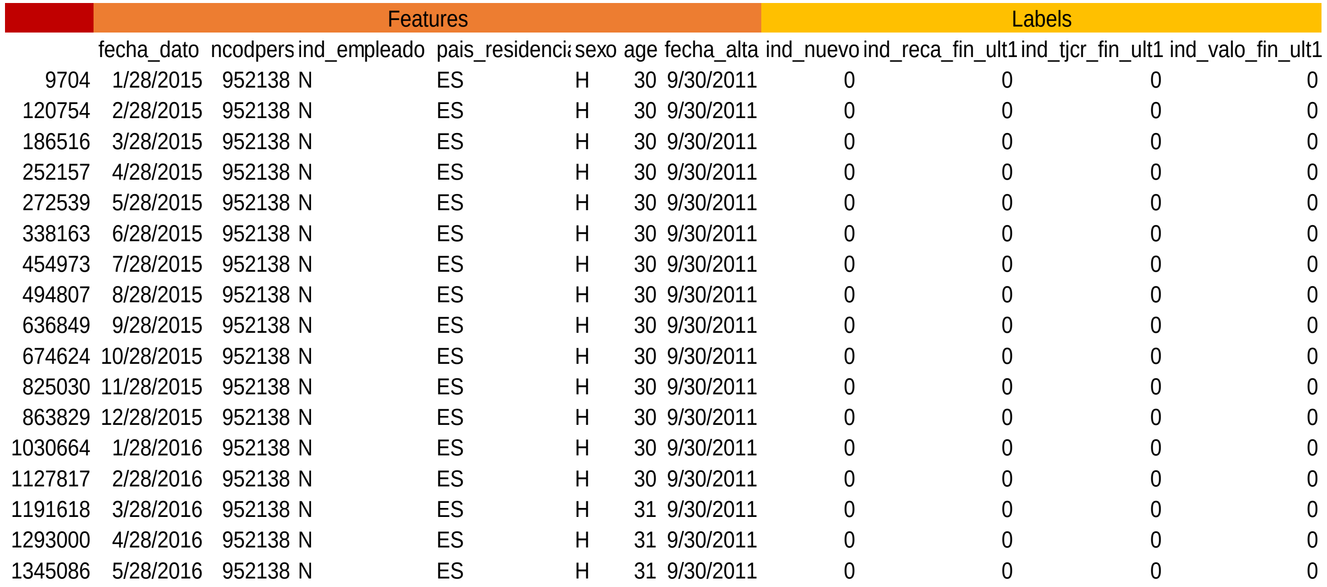
- **Our Goal:**

Based on customers' past behaviors and those of similar customers,

to **predict which products** their existing customers will use **in the next month.**

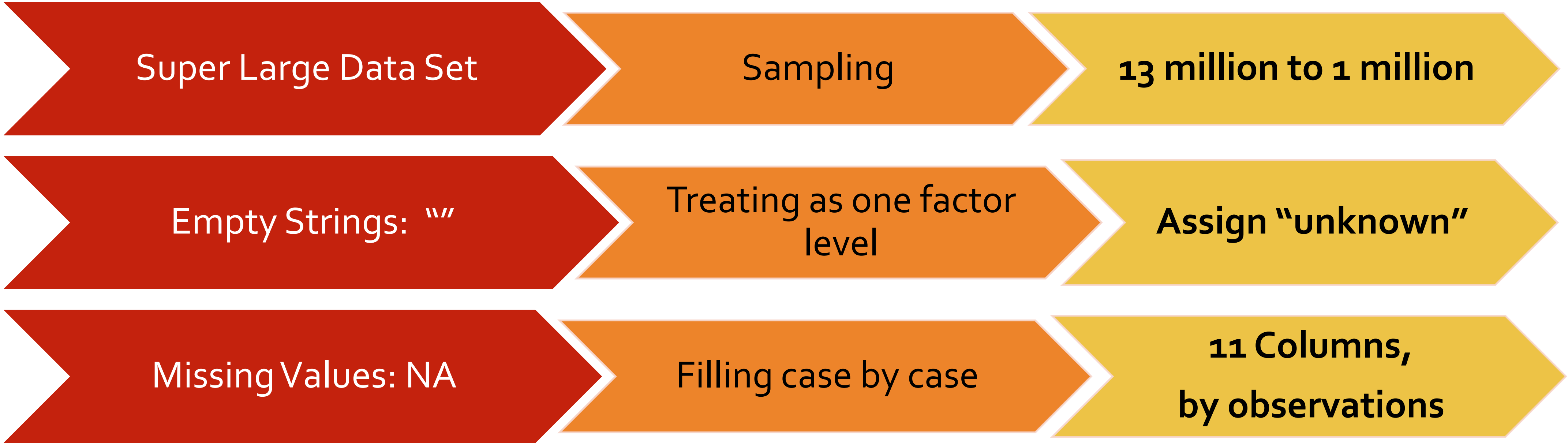
- **The Data**

  13 million x (24 features + 24 labels)



| | Features | | | | | | | Labels | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | fecha_dato | ncodpers | ind_empleado | pais_residenci | sexo | age | fecha_alta | ind_nuevo | ind_reca_fin_ult1 | ind_tjcr_fin_ult1 | ind_valo_fin_ult1 |
| 9704 | 1/28/2015 | 952138 | N | ES | H | 30 | 9/30/2011 | 0 | 0 | 0 | 0 |
| 120754 | 2/28/2015 | 952138 | N | ES | H | 30 | 9/30/2011 | 0 | 0 | 0 | 0 |
| 186516 | 3/28/2015 | 952138 | N | ES | H | 30 | 9/30/2011 | 0 | 0 | 0 | 0 |
| 252157 | 4/28/2015 | 952138 | N | ES | H | 30 | 9/30/2011 | 0 | 0 | 0 | 0 |
| 272539 | 5/28/2015 | 952138 | N | ES | H | 30 | 9/30/2011 | 0 | 0 | 0 | 0 |
| 338163 | 6/28/2015 | 952138 | N | ES | H | 30 | 9/30/2011 | 0 | 0 | 0 | 0 |
| 454973 | 7/28/2015 | 952138 | N | ES | H | 30 | 9/30/2011 | 0 | 0 | 0 | 0 |
| 494807 | 8/28/2015 | 952138 | N | ES | H | 30 | 9/30/2011 | 0 | 0 | 0 | 0 |
| 636849 | 9/28/2015 | 952138 | N | ES | H | 30 | 9/30/2011 | 0 | 0 | 0 | 0 |
| 674624 | 10/28/2015 | 952138 | N | ES | H | 30 | 9/30/2011 | 0 | 0 | 0 | 0 |
| 825030 | 11/28/2015 | 952138 | N | ES | H | 30 | 9/30/2011 | 0 | 0 | 0 | 0 |
| 863829 | 12/28/2015 | 952138 | N | ES | H | 30 | 9/30/2011 | 0 | 0 | 0 | 0 |
| 1030664 | 1/28/2016 | 952138 | N | ES | H | 30 | 9/30/2011 | 0 | 0 | 0 | 0 |
| 1127817 | 2/28/2016 | 952138 | N | ES | H | 30 | 9/30/2011 | 0 | 0 | 0 | 0 |
| 1191618 | 3/28/2016 | 952138 | N | ES | H | 31 | 9/30/2011 | 0 | 0 | 0 | 0 |
| 1293000 | 4/28/2016 | 952138 | N | ES | H | 31 | 9/30/2011 | 0 | 0 | 0 | 0 |
| 1345086 | 5/28/2016 | 952138 | N | ES | H | 31 | 9/30/2011 | 0 | 0 | 0 | 0 |

# DATA PROCESSING & CLEANING



Super Large Data Set → Sampling → **13 million to 1 million**

Empty Strings: “” → Treating as one factor level → **Assign “unknown”**

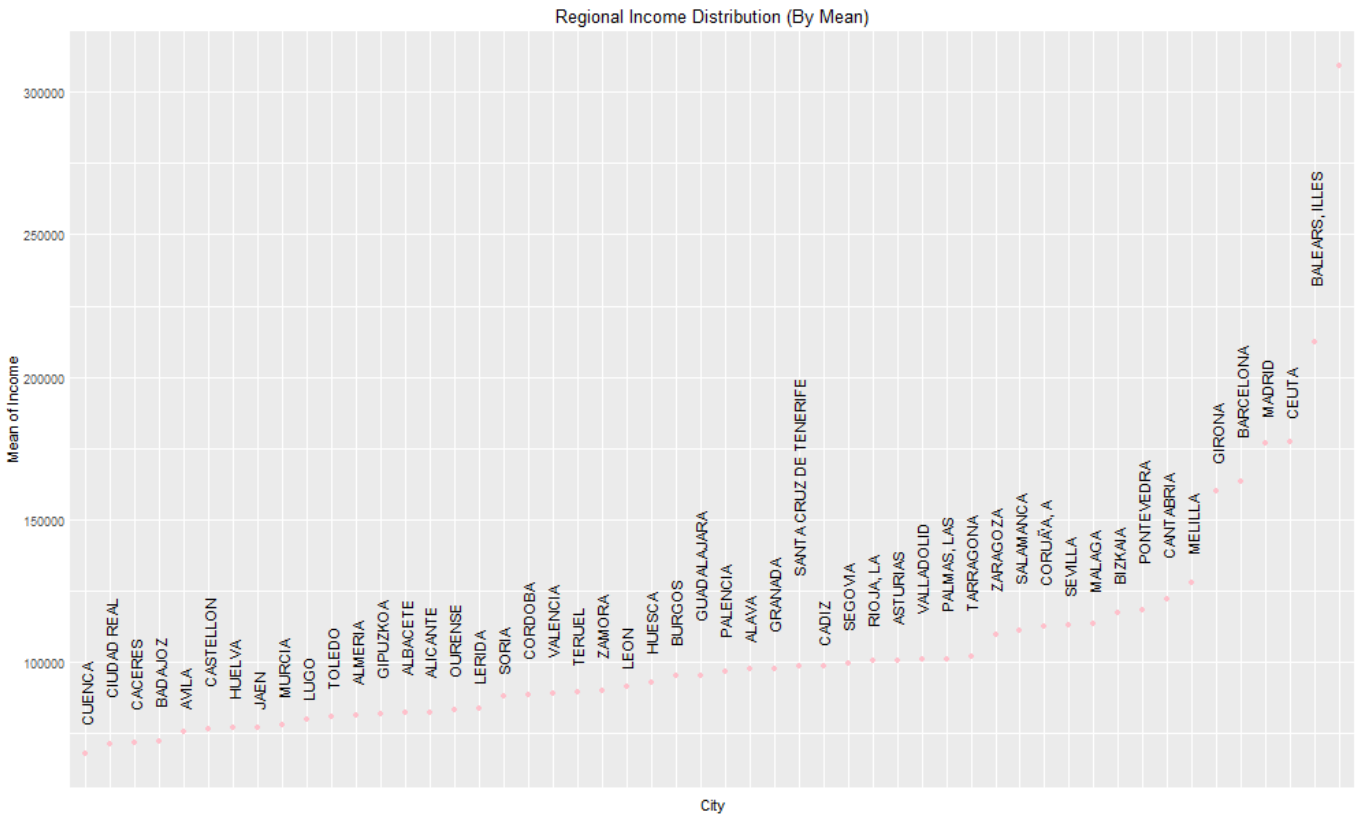
Missing Values: NA → Filling case by case → **11 Columns, by observations**

# FILLING IN MISSING VALUES FOR EXAMPLE: INCOME

## HOW TO

1. BY TOTAL AVERAGE?

**2. BY CITIES!**



# FEATURE ENGINEERING

- **Does time matter?**

  e.g. created “month’ feature (Christmas would buy more?).

  Cross-validation error rate from **9% to 10%** in the same model → abandoned

- **How to create features to reflect past behaviors?**

  Thoughts: can current situations be used to predict future (martingale)?

  Solutions: Appending current month’s labels to original features

  24 more features for each data

| New Features | | New Labels |
|---|---|---|
| April’s Features | April’s Labels | May’s Labels |

# MODEL SELECTION

- **What is the problem?**

  Multi-label Classification

- **Our Models (what we can do in R)**
  1. **Baseline ( = clustering?)**
  2. **rFerns in mlR**
  3. **XGBoosting**
  4. **Random Forrest**
  5. **SVM**

- **Not in R: ML-KNN, Neural Networks.**

# BASELINE MODEL

**1. Method**

**(1) Only Feature: customer ID** (ignore all other features, e.g. cities, age, income.)

Just consider the past purchasing behaviors of a certain customer A

**(2) Assign Predicated Value = Majority Label**

Calculated means of each label column for customer A

If P(A purchases product 1) > 0.5, then predict yes (assign label value = 1)

If P(A purchases product 1) <= 0.5, then predict yes (assign label value = 0)

**2. Cross Validation Error Rate = 0.87% (K=5)**

**3. Pros & Cons**

| Advantages | Disadvantages |
|---|---|
| no brainer.<br>easy to understand and compute.<br>Great performance | not using most features,<br>in a sense similar to guessing |

# Multi-label Classification in R
# {mlR} rFerns

**1. Method**

**{mlR} Machine Learning in R**, came out in Oct 2016

- Problem transformation methods (transform into binary/multiclass classification)
- Algorithm adaptation method (adapt multiclass algorithms so they can be applied directly)

**2. Cross Validation Error Rate = 24%**

| K=5 | Regular Method | Regular Method<br>+ Added Features |
|---|---|---|
| Error Rate | 24.00% | 30.18% |

**3. Pros & Cons**

| Advantages | Disadvantages |
|---|---|
| predicting all labels at the same time. | demanding requirements on the format of the data, e.g. Labels = logicals |
| convenient & neat technique. | not very good results |

# XG BOOSTING

## 1. Method

predicting labels column by column

## 2. Error Rate = 0.3%

| K=5 | Regular Method | Regular Method<br>+ Added Features |
|---|---|---|
| Error Rate | 4.12% | 0.29% |

## 3. Pros & Cons

| Advantages | Disadvantages |
|---|---|
| Great Performance | doesn't not take into account any of the prior months – strong assumption on the data! |
| Simply Implementation | |

# RANDOM FOREST

**1. Method**

Predicting labels column by column

**2. Error Rate =**

| K=5 | Regular Method | Regular Method<br>+ Added Features |
|---|---|---|
| Error Rate | 4.09% | 3.81% |

**3. Pros & Cons**

| Advantages | Disadvantages |
|---|---|
| enhance the strength of the model through averaging the results | Not very consistent outcomes among labels. 99.99% Vs 74.56% |
| Good results | Improvement not enhanced by a lot |

# SVM

## 1. Method

Predicting labels column by column

## 2. Error Rate = 4.66%

Tuning parameter (radial kernel, default gamma, cost = 0.001)

## 3. Pros & Cons

| Advantages | Disadvantages |
|---|---|
| Widely used model, stable | Hard to tune<br>Takes way too long time |
| easy to interpret | Once can fit only one<br>Column of labels |
| Good results | |

# CONCLUSIONS

**1. Feature Selection**

- Appending labels as new features is working!
- Sometimes less is more (baseline)!

**2. Model Selection**

- Direct Multi-label  Classification in {mlR} doesn’t work too well here.
- XG boosting is the best

**2. How to choose in the end?**

- Baseline: stable
- New Features: strong assumptions. Works in the short-run.
- XG Boosting + New Features is our selected model

# FUTURE CONSIDERATIONS

- The main objective of the project: **predict additional products next month**

  The error rate reflects which products will be owned,

  not which products were recently acquired

- **Feature Engineering**

  hard to incorporate past behaviors into account

  append baseline labels as features?

- **Combined Model**

  by model votes and majority labels?

- **R Shiny?**

# REFERENCES

- https://mlr-org.github.io/mlr-tutorial/release/html/multilabel/index.html#predict
- https://en.wikipedia.org/wiki/Multi-label_classification
- https://www.kaggle.com/c/santander-product-recommendation
- https://cran.r-project.org/web/packages/MLPUGS/vignettes/tutorial.html
- And many more!

# THE END
# THANK YOU!