Project Requirement

Due date: see syllabus section **Tentative Course Outline**

Submission process. Each team should submit on Blackboard a single compressed (i.e., zip) package containing two files: (1) a Jupyter Notebook file; (2) a Scored dataset (in .CSV format) containing only three columns—`id`, `probability`, and `classification` for those customers in the Score data. Build your deliverable (report) content in the Jupyter notebook file: your codes will be built in the Code cells (run all codes); your report content will be built in the Markdown cells and should include an introduction, project objective, process summary, comments on any relevant EDAs, rationale for each major project decision, results and interpretation, final conclusion, and recommendations.

1. Introduction

This project will give you an opportunity to apply most of the methods and techniques covered in class. Feel free to go beyond what is covered, and experiment with new models and algorithms if you so desire! After all, you will be competing with other teams, and your prediction results will be compared with others and ranked in a leaderboard! In your project, your team will have a chance to showcase and further sharpen skills on choosing, building, evaluating, and improving supervised machine-learning models in Python, in the context of a real business problem. Your team will need to apply independent judgment and make choices, some of which will be tough; your judgment should be informed by the nature of the problem and the specific business case, as well as the exploratory data analyses (EDA) that you conduct on the dataset.

Objective

Your team will predict the probability of purchase outcome of the customers in the Score data. These customers are contacted by the sales representatives of the Company (anonymous) to consider buying a car insurance product. Your classification results ("purchased" or "no purchase") are based on your predicted probabilities. Build a range of models. Using open-source packages (e.g., `sklearn`, `keras`, `tensorflow`) is highly recommended. The data you will use are described below.

2. Background and the Goal

Data are provided by a large International Insurance company. Company offers a range of insurance products, e.g. medical insurance, life insurance, homeowners insurance, car insurance, etc. Company routinely cross-sells its insurance products. This dataset records Company's attempts to cross-sell car insurance to its current medical insurance policyholders who are believed to be vehicle(s) owners (based on information collected about policyholders). If there is a need, current policyholders may have an incentive to purchase additional insurance products from the same company, such as potential discount applied to the multi-policy annual premium. In most countries and with few exceptions (e.g., life insurance is an exception), customers are able to switch provider or terminate a

policy anytime without incurring any significant penalty. Description about the datasets and columns can be found in the Data section.

> *Good to know* –
> An insurance policy – also called a contract of adhesion – is an agreement between a person and his/her insurer which outlines the coverage the insurer will provide to the insured. Under the specification of the policy, insurance provides guaranteed compensation for specified loss, damage, illness, or death in return for a specified premium. An insurance premium refers to how much an insurance policy costs, typically on an annual or semi-annual basis.

3. Project Performance Evaluation

Your final project performance will be determined by two parts: (1) quality of prediction as reflected in two measures: AUC ROC (a.k.a. ROC AUC, or Area Under the Curve)[1], and F-1 score; (2) quality of final project deliverable, whose evaluation criteria and rubric are described below.

Criteria of Final Project Deliverable

Build your deliverable contents in the Markdown cells of the main Jupyter notebook file.

Your deliverable contents should clearly and concisely document the procedures, rationales, and findings of the procedures you employ in fulfilling the objective of the project. Your deliverable should at least have three main sections: (1) a Data Preparation section including, but not limited to, the procedures undertaken to import the dataset, train-test split and evaluate the validity of the split, and prepare the final Train and Test sets; (2) a (brief) Exploratory Data Analyses section including, but not limited to, any descriptive summary, data manipulation and transformation, possible outlier and erroneous value diagnosis, and any visualization that can support the decision of properly implementing, tuning, and evaluating the modeling algorithm(s); note, analysis of missing value is not necessary; also, Section (1) and (2) may not have a clear-cut boundary, so feel free to format those sections deemed appropriate for your project; (3) Model Development and Evaluation section, including the procedures used to choose, train, and evaluate any modeling algorithms, judgments related to the decision analysis of the business problem as relevant to model evaluation, and those procedures used to produce the final Scored Dataset containing only three columns: `id`, `probability`, and `classification`. Do not forget to export this Scored data frame as a .CSV file and submit with your notebook file.

Tips on Improving Project Performance

Be prepared to go beyond materials and methods covered in the class / homework / exam, but use the materials and discussions from the class as a guide, rather than "going wild" in your (or your team's) exploration. There is nothing wrong about being exploratory, but be mindful about the

[1] This is why you should include your predicted probabilities in Score dataset, as requested. For the probabilities, only include the predicted probabilities for the "purchased" outcome; do not include the predicted probabilities for the "no purchase" outcome even if you obtained these results!

feasibility – how much time do you or your team have in exploring this approach? Will you be able to wrap up, train model, obtain and evaluate results? Will this "wild chase" of a "cool model" affect the basic implementation and completion of your project?

That said, solely using the materials covered by the homework / exam will typically not be sufficient for you to get good results. One example is that using state-of-the-art open-source packages for building artificial neural networks (e.g., `sklearn`, `keras`, `tensorflow`) is discussed only in class demo as a part of self-experimentation, but not tested in any homework assignment. I will strongly recommend you to at least understand what is covered in those hands-on demo and learn to use and experiment with the basic models provided by any of these packages. From there, it's up to you and your team if you would like to self-learn even more tools and methods to improve your solution.

10-fold or 5-fold cross-validation is strongly recommended. Grid search may be helpful to optimize your model specification, but keep in mind that doing grid search too broadly (e.g., exhausting all possible combinations of all hyper-parameters) will not provide a good enough marginal benefit; in other words, too much time can be spent on efforts that would not necessarily improve your model, or that may only improve performance by a little bit.

4. Data

You are given a large training dataset which includes the target column – whether a customer purchased or did not purchase the car insurance being sold. Columns are listed below.

You are also given a Score dataset that consists of all of the columns in the training dataset, except the target column. Columns are listed below.

Training dataset "bzan6357_insurance_3_TRAINING.csv"

id	(`str type`) 9-letter unique customer ID
buy	(`num type; target`) cross-selling outcome by 30 days post-contact, 1 = "purchased vehicle insurance", 0 = "no purchase"
age	(`num type`) customer's age at the time of contact
gender	(`str type`) "male" or "female"
tenure	(`num type`) number of days since policyholder's current medical insurance has started
region	(`num type`) unique code-number assigned to the region in which customer lives (caution! many discrete values!)
dl	(`num type`) 1 = "has valid driver license", 0 = "no driver license"
has_v_insurance	(`num type`) 1 = "already has valid car insurance", 0 = "no valid car insurance"
v_age	(`str type`) vehicle age, three possible labels: "1-2 year", "< 1 year", "> 2 years"
v_accident	(`str type`) "yes" or "no", vehicle had accident(s) before
v_prem_quote	(`num type`) annual premium quote for the cross-sold car insurance (local dollar, possibly different for each policyholder)
cs_rep	(`num type`) unique code-number assigned to the customer-service representative who attempted to cross-sell the car insurance (caution! many discrete values!)

Score dataset "bzan6357_insurance_3_SCORE.csv": same as Training dataset except column "buy"