Rishi

# Scenarios

小王开了一家淘宝店，双十一快要到了，他想知道自己要通知供货商准备多少货。他细心的记录了之前每一年双十一期间的销量，并决定试一试AICloud提供的时间序列预测工具。他将往年销量数据输入了Excel表格中，点击预测很快得到了未来一段时间的包裹数量。他看到了很多个不同的预测结果和变化曲线，他选择让工具自动为他挑选出了最保守的一个结果，并按照结果补充了存货。

小李是一名Azure运维人员。国庆假期快要到了，小李却安排了on-call。为了应对国庆期间可能出现的突发情况，小李准备做一下 Forecasting。他以前手动做过相关工作，不过需要自己找模型调参，他厌烦了这样的事情，便打开了 AICloud 团队提供的时间序列预测工具。他上传了九月服务器 PeakDailyCores 数据，并迅速获得了未来七天的预测结果。他还顺便上传了自己以前训练的模型，发现在验证集上 Rishi 表现得好很多，于是美滋滋的把自己的模型丢尽了垃圾桶。为了结合小李的经验，他对工具内置的模型和一些超参进行了修改，并通过AICloud的工具迅速进行了调参和模型融合，并对可能出现的情况进行了准备。小李美滋滋地下班回家了。

# Features

* From the User Scenario Perspective
  + 用户友好的界面，无论对专家还是无经验者
  + 自定义 loss function，目标函数。自动调参
  + 整合数据库查询语言，例如 Kusto
* From the Model Integration Perspective
  + 使用现成的模型，在提供的验证集上选择最佳的模型
  + 在数据量的基础上选择模型（基于模型的先验知识）
  + 允许自定义模型（满足特定格式），并进行 fair comparison

# Goals & Non-Goals

## Goals

* 在 state-of-art 模型的基础上，提供一个容易使用的工具
* 为用户找到最优的模型和参数
* 整合进入 Azure

## Non-Goals

* 我们不会设计新的算法和模型

# UX Design

* For Non-Experts
  + 通过简单的 drag-and-click 操作完成输入
  + 如果集成到 Azure 中，由于服务器数据已在服务器中，可以通过选择时间界限来完成输入
  + 默默完成繁杂的工作，而不用困扰用户
    - 通过几次点击即可完成数据预处理
    - 自动调参
  + 通过可视化的方式将成果呈现给用户（绘图）
* For Experts
  + 通过数据库查询脚本完成输入（如 Kusto）
  + 用户自定义损失函数
  + 上载固定格式的模型

# Competitors

* Prophet，<https://facebook.github.io/prophet/>，一个时间序列预测的工具包，有 R 和 Python 接口。在节日相关的数据影响上表现良好。
* Azure ML，<https://azure.microsoft.com/en-us/services/machine-learning/>。一个用户友好的深度学习工具，主要提供 tree-based 模型。
* GluonTS，<https://gluon-ts.mxnet.io/> 和 Prophet 类似，提供时间序列预测的工具包。相对 Prophet 更加灵活，但部分功能需要通过自己继承类来实现。

# Plan

|  |  |
| --- | --- |
| Time | Work |
| Week2 | Feature Spec |
| Week4 | Design Spec |
| Week9 | Code complete |
| Week10 | ZBB (Zero Bug Bounce) |
| Week12 | Release |

## Proposed solution

* User interface
* Input module
* Data pre-processing module
* Model execution and parameter search module
* Output module