# 选型测试：AutoGluon

测试人:牟童

日期：2022年05月20日

1. 产品概述
   1. 产品简介

AutoGluon 是一个亚马逊研发的，开源的、易使用、易扩展的AutoML自动化机器学习框架，擅长于解决生活中的实际问题（包含图片，文字，表格型数据），无论是机器学习初学者还是机器学习专家，都可以得到以下帮助：

* 可以使用原始数据快速构建（几行代码）深度学习和传统机器学习模型
* 不需要机器学习专家的帮助，就可自动选择最先进的技术构建模型
* 利用自动超参数搜索技术、自动模型选择技术、自动数据处理等技术
* 可以根据你的需求，轻松改进或调整定制的模型
  1. 产品功能

AutoGluon擅长解决图片（image）、文字（text）、表格型（tabular）三种类型数据，根据业务场景，官方文档罗列了5种常见的任务，并列出了简单教程，分别是：

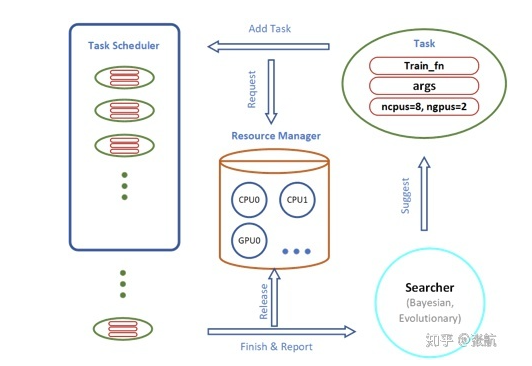
* 表格数据预测（Tabular Prediction）：解决的问题是如何预测表格数据，文档链接：<https://auto.gluon.ai/stable/tutorials/tabular_prediction/index.html>
* 图片数据预测（Image Prediction）：解决的问题是图片分类任务，文档链接：<https://auto.gluon.ai/stable/tutorials/image_prediction/index.html>
* 目标检测（Object Detection）：解决的问题是图片目标检测任务，文档链接：<https://auto.gluon.ai/stable/tutorials/object_detection/index.html>
* 文字数据预测（Text Prediction）：解决的问题是自然语言处理（NLP）相关问题，文档链接：<https://auto.gluon.ai/stable/tutorials/text_prediction/index.html>
* 多模态数据预测（Multimodal Prediction）：解决的问题是如何同时处理包含图片、文字、数字等多模态的数据，文档链接：

<https://auto.gluon.ai/stable/tutorials/tabular_prediction/tabular-multimodal.html>

* 1. 安装环境
* 支持在linux、mac、windoes平台的下载和安装
* 支持github克隆或pip下载
* 支持CPU和GPU版本
* 支持Python版本：3.7、3.8、3.9
* 用户可根据自己系统的配置选择对应的下载命令，下载地址如下：https://auto.gluon.ai/stable/index.html

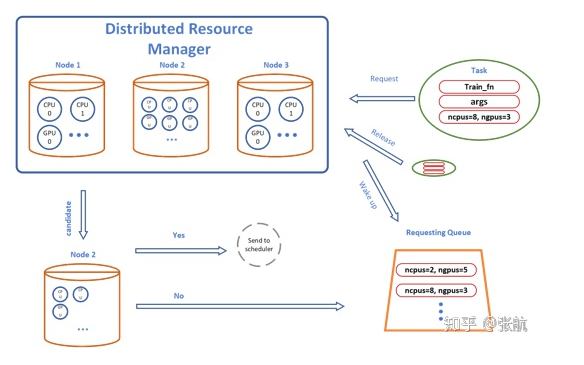
1. 产品架构
   1. 任务系统架构

系统里面每一部分都是模块化的，便于扩展。每一个任务都是由主函数、参数、计算资源三部分组成的。其中搜索算法（Searcher）可以提供参数优化，资源管理器（Resource Manager）动态管理分发计算资源，任务调度器（Task Schedule）来管理任务调度和提前终止，下面是任务系统架构图。



* 1. 分布式资源管理

AutoGluon另外的一个优势是分布式训练，不管开发者有没有分布式训练的经验，只需要提供一个IP地址，AutoGluon会自动地把任务分发到远端机器上，后台会自动的根据不同机器的计算资源来自动调配，下面是分布式的资源管理流程图。



* 1. 特点总结

AutoGluon主要应用于三大领域，具有两大调参功能。

三大领域主要包含图片、文本、表格型数据，当然还包含三者结合的多模态数据（Multimodal Prediction）：

* Image：图片分类(image classification)、目标检测(object detection)
* Text：文本分类（text classification）
* Tabular data：表格数据预测（tabular prediction）

两大调参功能：

* 自动调参：框架用mxnet写的，也支持Pytorch，支持的搜索策略包含random search、grid search、RL、Bayesian optimization等
* NAS（神经架构搜索）：仅支持image类型数据，目前只有ENAS。
  1. 解决的业务痛点

在现实的机器学习模型构建中，有多个痛点需要解决：

* 构建机器学习模型环节多，流程长
* 依赖数据科学、机器学习专家的知识
* 模型难以通用
* 专业人才数量缺乏

其次，在实际业务中也存在越来越明显的诉求：

* 缩短AI项目的投产时间，降低建模所需人员规模和专业能力的门槛，通过自动化技术将AI模型的构建缩短至数天甚至数小时。
* 降低AI项目的建设成本，通过标准化的建模过程，实现AI的规模化应用，减少AI落地的总成本。
* 支撑业务的快速变化，利用高性能的AI开发工具实现模型的快速迭代，灵活适应业务变化。

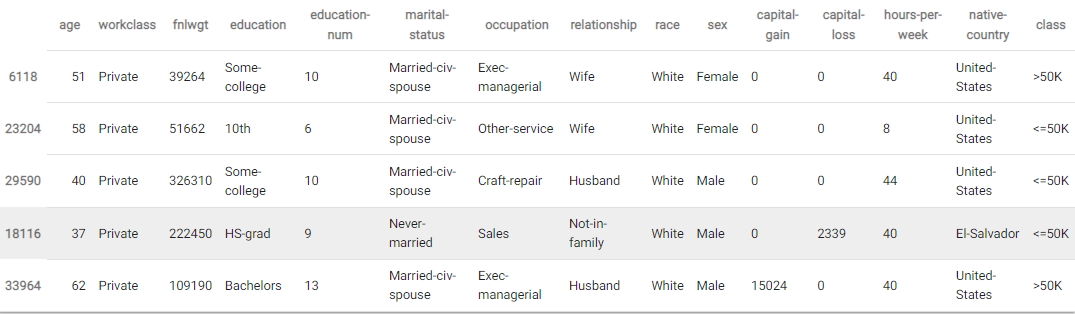
AutoGluon的核心目标就是用更低的计算开销，输出更改的泛化能力模型。AutoGluon就可以帮助企业用更低的成本得到更高效的目标。

1. 测试实例

接下来用，表格数据预测任务（Tabular Prediction）来演示如何使用AutoGluon框架。

任务描述：构建一个分类模型来预测一个人的年收入是否超过$50000。

数据描述：我们使用AutoGluon提供的默认数据集，该数据集描述了一个人的特征，比如教育情况，工作地点，职业，性别，种族，国家等等，示例如下;



* 1. 导入数据
* 首先导入必要的库

1. **from** autogluon.tabular **import** TabularDataset, TabularPredictor

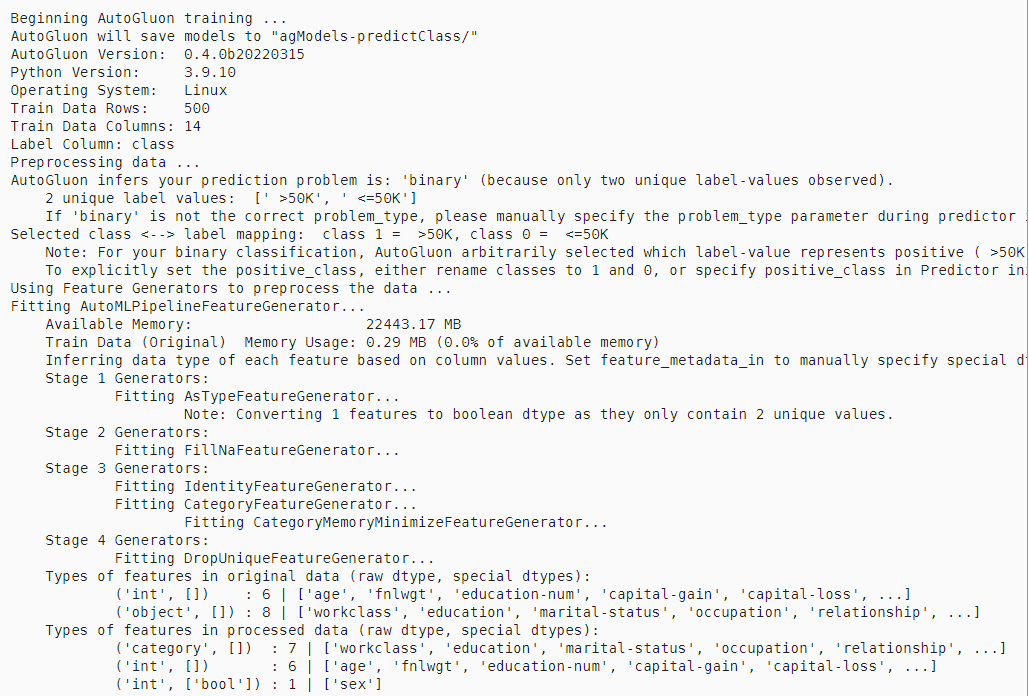
* 导入数据

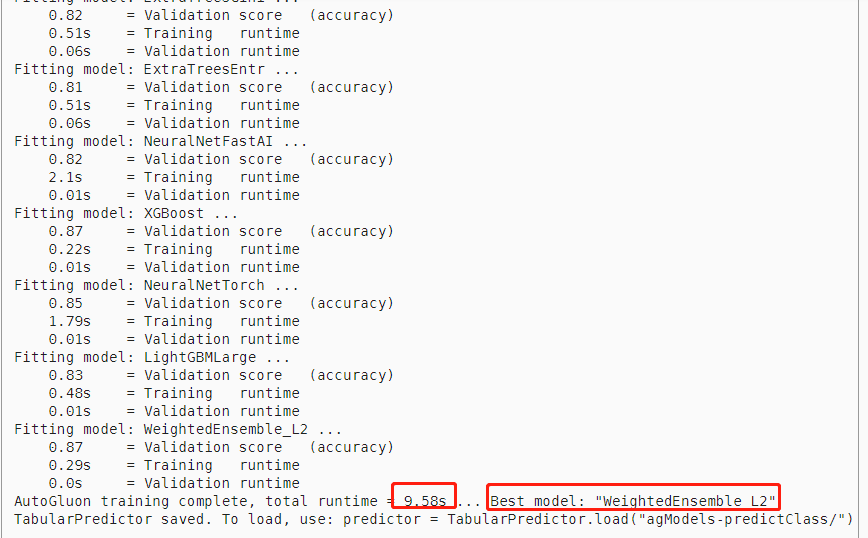
1. train\_data = TabularDataset('https://autogluon.s3.amazonaws.com/datasets/Inc/train.csv')
2. subsample\_size = 500
3. train\_data = train\_data.sample(n=subsample\_size, random\_state=0)
   1. 训练模型

* 进行训练模型

1. save\_path = 'agModels-predictClass'
2. predictor = TabularPredictor(label=label, path=save\_path).fit(train\_data)

* 训练日志显示如下



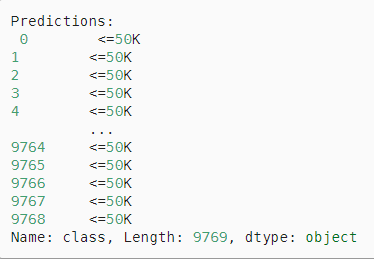


* 1. 预测数据（最好的模型）
* 导入测试数据

1. test\_data = TabularDataset('https://autogluon.s3.amazonaws.com/datasets/Inc/test.csv')
2. y\_test = test\_data[label]
3. test\_data\_nolab = test\_data.drop(columns=[label])

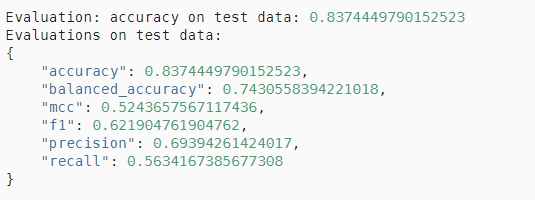
* 进行预测，并显示预测结果

1. predictor = TabularPredictor.load(save\_path)
2. y\_pred = predictor.predict(test\_data\_nolab)
3. **print**("Predictions:  \n", y\_pred)



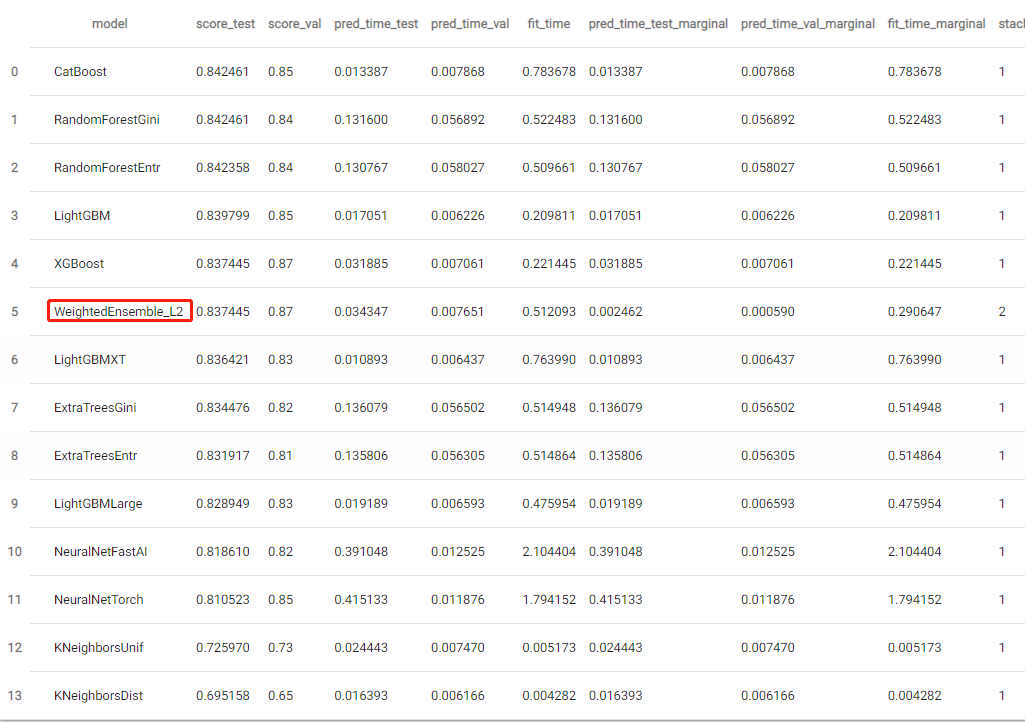
* 得出评价指标metrics

1. perf = predictor.evaluate\_predictions(y\_true=y\_test, y\_pred=y\_pred, auxiliary\_metrics=True)



* 1. 多模型结果
* 也可以显示多模型的预测结果

1. predictor.leaderboard(test\_data, silent=True)



通过以上简单的几行代码，即可快速的构建机器学习模型，并且AutoGluon会自动尝试不同的模型，并聚合出一个综合的模型，具有很强的泛化能力，可以帮助算法工程师快速构建模型，没有必要花费大量时间在调参、选择适合的算法模型上了。