# 纽约出租车费预测
# 实验报告 PPT

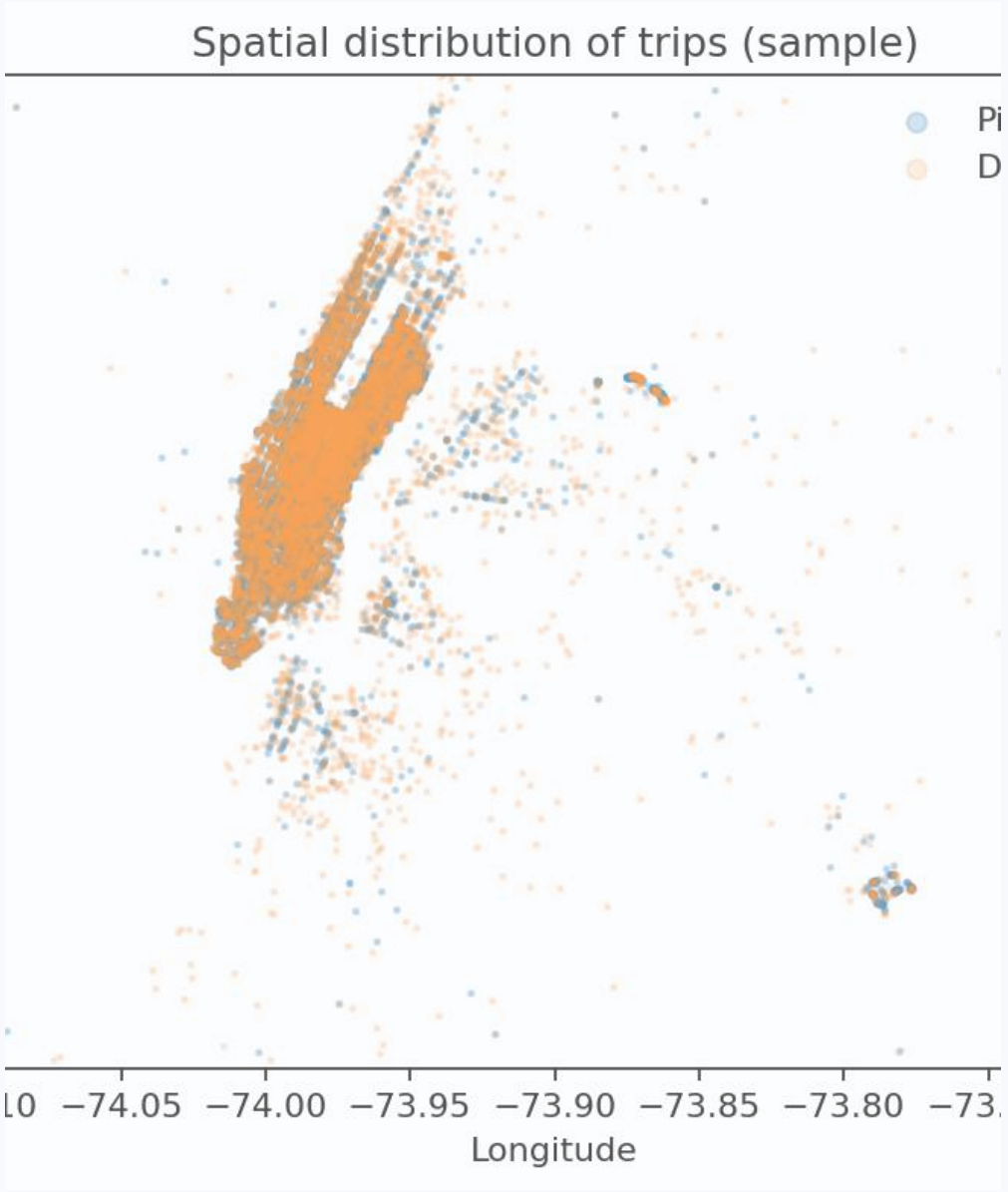
依据你提供的文章思路，并结合上传的 train.csv 样本（50,000 行）完成实验设计、建模与结果汇报。

**50,000**
原始样本行数

**48,770**
清洗后样本

**3.87**
随机森林验证 RMSE

*实验目标：根据时间、起终点经纬度和乘客数预测车费 fare_amount*



# 1. 实验目标与数据概况

先明确任务、字段和整体实验路线

任务定义

回归预测：输入行程的时空信息与乘客数，输出出租车费用 fare_amount。

7

原始字段数

2009 - 2015

时间覆盖

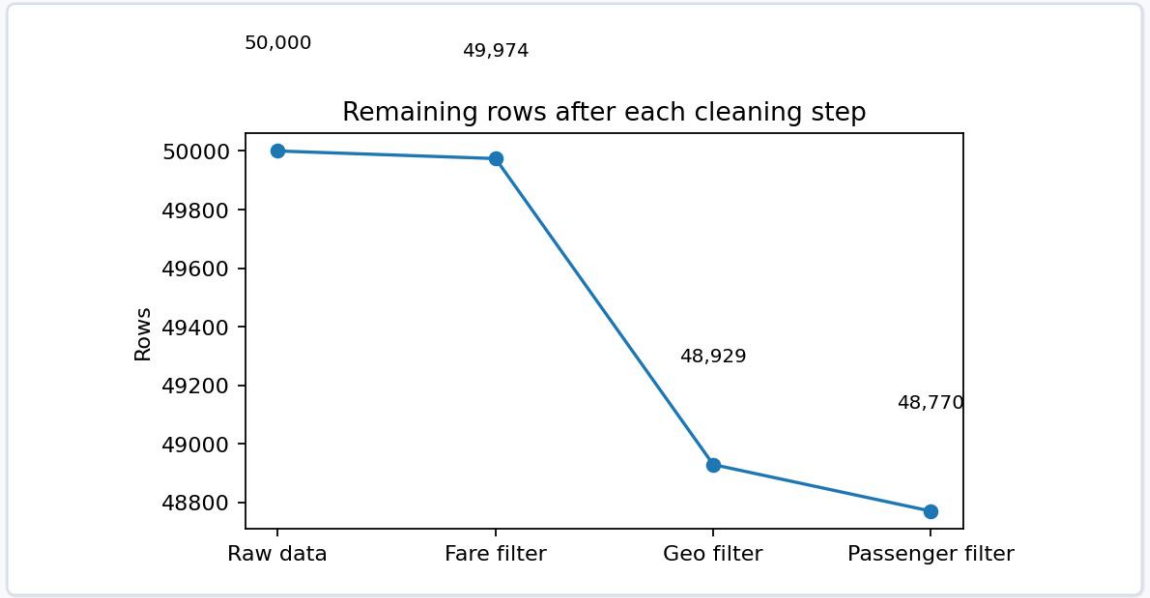
$11.33

清洗后均值票价

字段说明

- 目标变量：fare_amount
- 时序字段：pickup_datetime
- 空间字段：起点 / 终点经纬度
- 离散字段：passenger_count

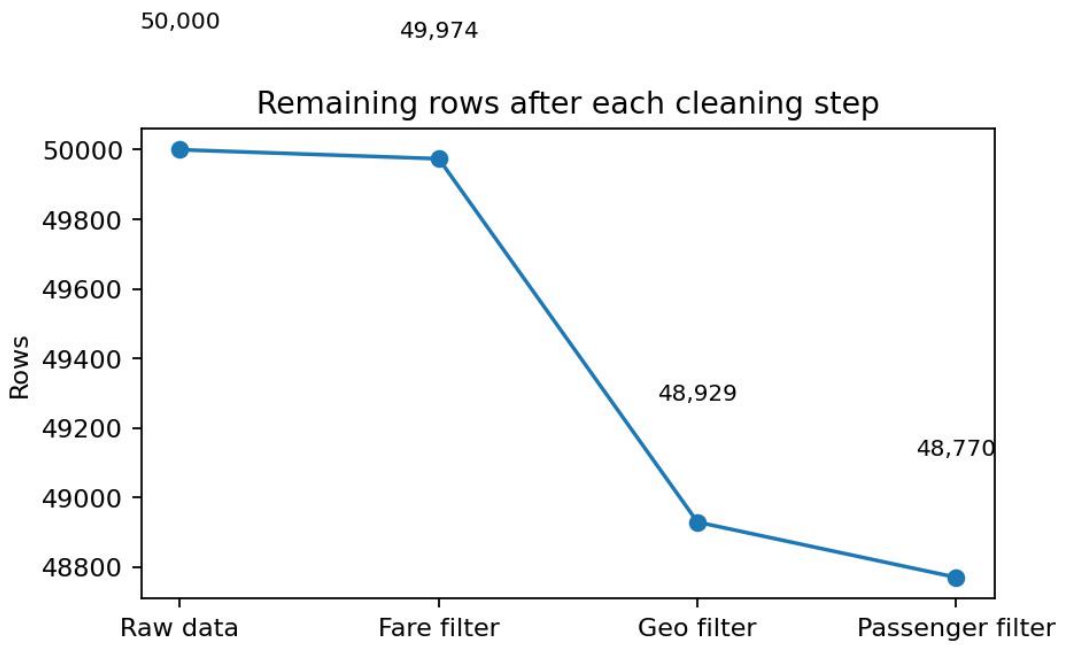
实验流程


上传数据为 50,000 行样本，不是文章中的 500 万行；因此本次结果更适合做课程实验与方法演示。

# 2. 数据清洗策略与结果

先做规则过滤，再进入特征工程和建模




清洗规则

## 费用异常

- 删除负费用 6 条、0 费用 3 条、>100 美元 16 条
- 保留区间：2.5 ⩽ fare_amount ⩽ 100

## 地理异常

- 按纽约区域限定经纬度：纬度 40 - 42，经度 -75 至 -72
- 该步骤后样本从 49,974 降至 48,929

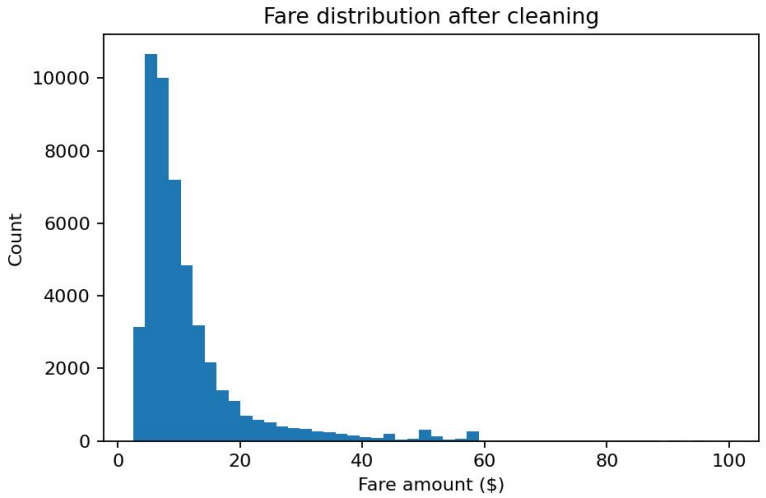
## 乘客数异常

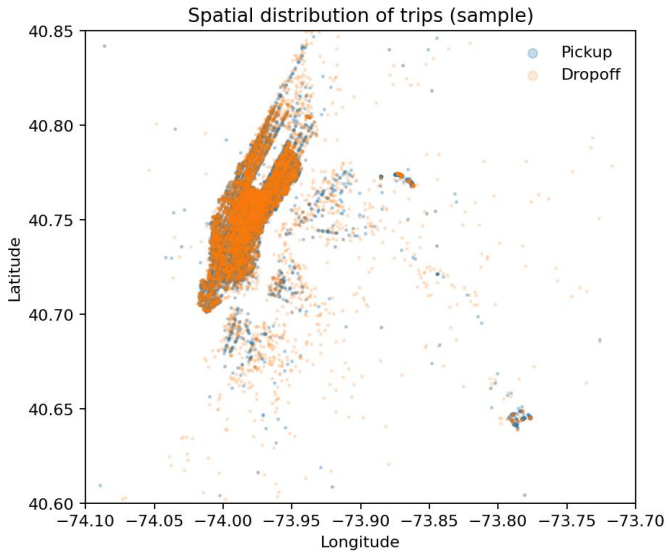
- 仅保留 1 - 6 人的正常订单
- 最终保留 48,770 行，占原始样本 97.5%

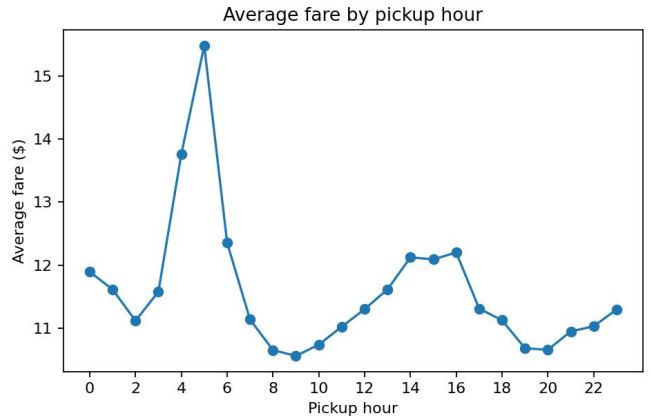
结论：上传样本总体质量较高，主要问题来自经纬度越界，其次是极端票价。

# 3. 探索性分析（EDA）

看分布、看空间、看时段，先理解数据再建模




**核心发现**

- 票价右偏，清洗后均值 $11.33，中位数 $8.5。
- 上车与下车热点明显集中在曼哈顿中城与机场走廊。
- 凌晨 4–6 点平均票价更高，最高时段为 4 点。

# 4. 特征工程设计

核心思路：把时空信息转成机器学习可直接学习的数值特征

特征家族

## 空间特征

- abs_lat_diff
- abs_lon_diff
- manhattan
- haversine

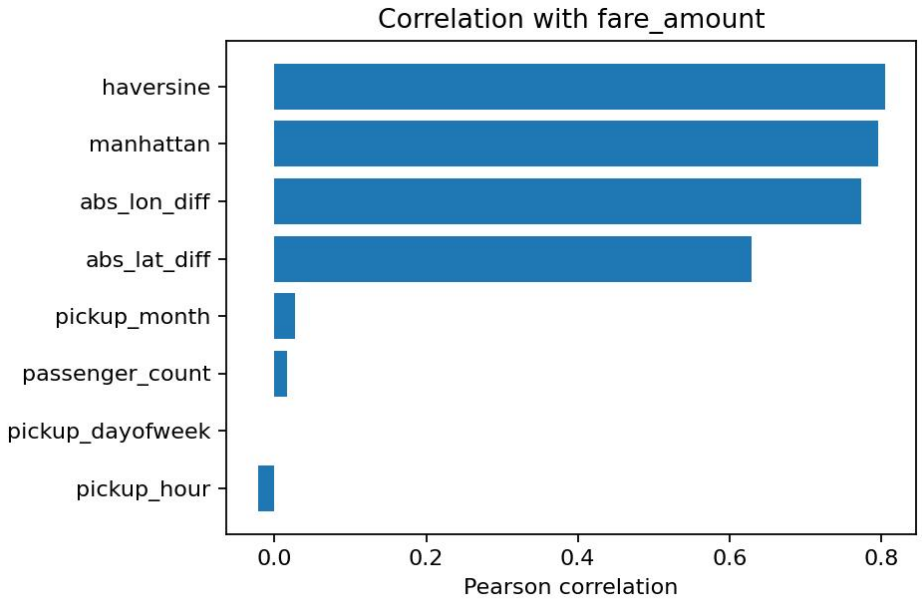
## 时间特征

- year / month / day
- dayofweek / hour
- frac_day / frac_week
- pickup_elapsed

## 基础特征

- pickup_latitude
- pickup_longitude
- dropoff_latitude
- passenger_count

重点公式：Haversine 距离用于估计地球表面两点间真实球面距离，是本实验最关键的特征。




相关性结论：Haversine 与票价相关系数约 0.80，明显高于时间类特征。

# 5. 实验设置与复现步骤

照着这一页即可把实验完整复现出来

### 实验协议

- 数据切分：fare_amount 分箱后做分层抽样，训练集 80%，验证集 20%。
- 评价指标：RMSE、MAE、MAPE、$R^2$；其中 RMSE 作为主指标。
- 基线模型：线性回归；对照模型：随机森林。
- 随机森林参数：n_estimators=80，max_depth=20，random_state=42。

### 复现流程

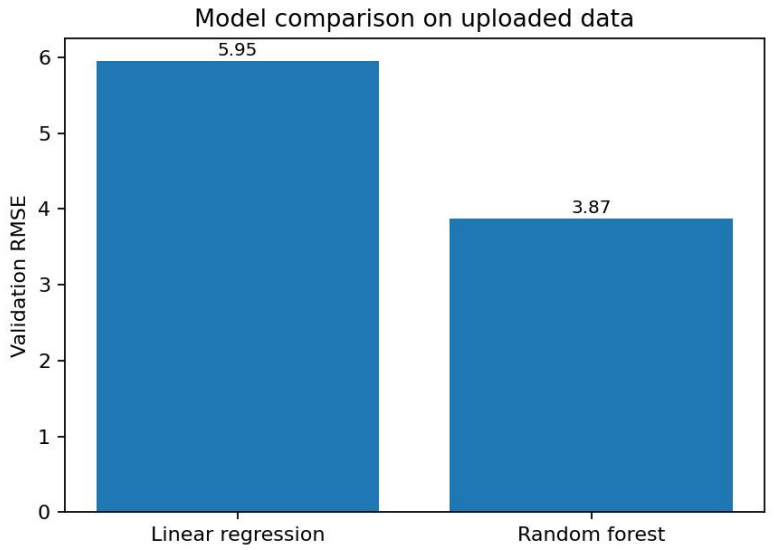
- 读取 train.csv，并 parse_dates=[pickup_datetime]
- 按票价、经纬度、乘客数做规则过滤
- 构造空间与时间特征
- train_test_split + stratify(fare_bin)
- 训练线性回归与随机森林
- 输出 RMSE / MAE / MAPE / $R^2$ 并画图

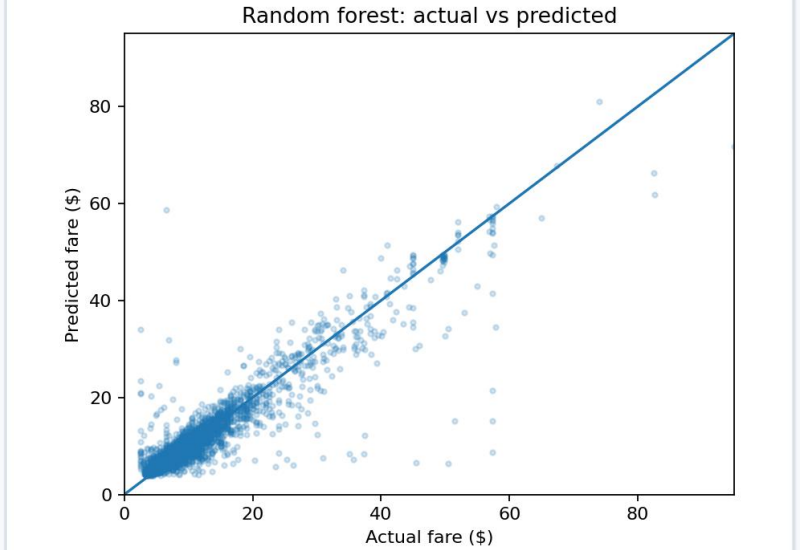
### 最小代码骨架

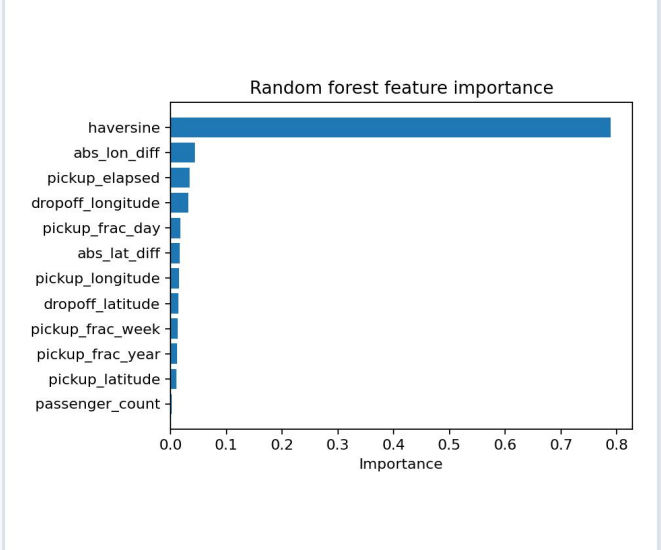
```
1 data = pd.read_csv('train.csv',
parse_dates=['pickup_datetime'])
2 data = clean_rules(data)
3 data = make_features(data)
4 X_tr, X_va, y_tr, y_va =
train_test_split(..., stratify=fare_bin)
5 lr.fit(X_tr[basic_features], y_tr)
6 rf.fit(X_tr[features], y_tr)
7 evaluate(lr, X_va); evaluate(rf, X_va)
```

# 6. 模型结果对比

随机森林明显优于线性回归，但也出现了一定过拟合




| 模型 | 验证 RMSE | 验证 MAE | 验证 MAPE | $R^2$ |
|---|---|---|---|---|
| 线性回归 | 5.95 | 2.88 | 30.1% | 0.598 |
| 随机森林 | **3.87** | **1.88** | **19.8%** | **0.830** |

随机森林验证 RMSE 从 5.95 降到 3.87，下降 35.0%。
训练集 RMSE 仅 1.49，说明模型已有一定过拟合，需要继续调参或换 boosting 模型。

# 7. 结论、答辩话术与后续优化

这一页可直接作为实验报告结尾

### 实验结论

- 空间距离是最强信号，Haversine 特征的重要性接近 0.79。
- 线性模型可做基线，但无法充分拟合票价的非线性关系。
- 随机森林在当前样本上取得更优表现，验证集 $R^2$ 达 0.830。

### 汇报建议

- 先讲数据清洗规则，再讲特征工程，最后讲模型提升幅度。
- 回答“为什么距离最重要”时，可直接引用相关性图和特征重要性图。
- 回答“为什么还不够好”时，指出随机森林已有过拟合，且未引入天气、道路距离等外部变量。

### 下一步优化

- 扩大样本：从 5 万行提升到 500 万行甚至全量。
- 换模型：LightGBM / XGBoost 通常比随机森林更强。
- 加特征：机场/商业区 POI、天气、道路距离、拥堵等级。
- 做验证：K-fold 或按时间切分，避免随机切分带来的乐观偏差。
- 做部署：保存模型、封装预测 API、监控线上 RMSE。

**一句话总结：这份实验报告最值得强调的是“规则清洗 + 时空特征 + 随机森林”，它构成了一个完整、可复现、适合课程展示的机器学习回归案例。**