**Airline Arrivals Project**

Use this [dataset of airline arrival information](http://stat-computing.org/dataexpo/2009/the-data.html) to predict how late flights will be. A flight only counts as late if it is more than 30 minutes late.

1. The project should follow guideline as previous projects.
2. Apply models in
   1. Naïve Bayes,
   2. Logistic Regression,
   3. Decision Tree,
   4. SVM.
3. Apply

PCA

1. Compare performances among models, write up analysis why the model is good or bad in the algorithmic approach (explain why the algorithm is good or bad for the dataset structure, can you do something to improve the model?)
2. Include the conclusions.

Dự án hàng không

Sử dụng bộ dữ liệu này về thông tin đến của hãng hàng không để dự đoán chuyến bay sẽ trễ như thế nào. Một chuyến bay chỉ được tính là trễ nếu trễ hơn 30 phút.

1. Dự án nên theo hướng dẫn như các dự án trước.

2. Áp dụng mô hình trong

a. Bay ngây thơ,

b. Hồi quy logistic,

c. Cây quyết định,

d. SVM.

3. Áp dụng

PCA

4. So sánh hiệu suất giữa các mô hình, viết phân tích tại sao mô hình tốt hay xấu trong cách tiếp cận thuật toán (giải thích tại sao thuật toán tốt hay xấu cho cấu trúc dữ liệu, bạn có thể làm gì để cải thiện mô hình không?)

5. Bao gồm các kết luận.

1

//////// From the graph above, which features, in your opinions, should be removed, which should be kept, why?

2

////// Run codes above, describe what you see and start clean data / remove unimportant columns.

3

/////// start your code below to one hot encode or label encoding categorical features

4

/////// \_normalize / scale your data to range [0,1]

5

//////////// the code below has "n" as dimensions to reduce to, decide that number so the total explation ratio > 0.95

6

////// Search google for "sklearn logistics regression" / decision tree / svm and figure out how to work on below parts

1

//////// Từ biểu đồ trên, những tính năng nào, theo ý kiến của bạn, cần được loại bỏ, những gì nên được giữ lại, tại sao?

2

////// Chạy mã ở trên, mô tả những gì bạn thấy và bắt đầu xóa dữ liệu / xóa các cột không quan trọng.

3

/////// bắt đầu mã của bạn bên dưới thành một tính năng phân loại mã hóa hoặc nhãn mã hóa nóng

4

/////// \_n normalize / scale data của bạn thành phạm vi [0,1]

5

//////////// mã bên dưới có "n" làm kích thước để giảm xuống, quyết định số đó để tổng tỷ lệ khám phá> 0,95

6

////// Tìm kiếm google cho "hồi quy hậu cần sklearn" / cây quyết định / svm và tìm ra cách làm việc trên các phần dưới đây

1

* To classify whether a flight is late or not, we choose **ArrTime as the label**.
* **ActualElapsedTime, CRSElapedTime, AirTime** and **Distance** have very high correlation (>0.95) so we can just keep one feature (Distance) as predictor.
* **CRSDepTime** and **DepTime** also has 0.97 correlation score, we just keep **DepTime** as it is the real time data.
* **CRSArrTime** and **ArrTime** also has 0.87 correlation score, we just keep **ArrTime** as it is the real time data.
* **ArrTime**, and **DepTime** although have high correlation (0.72) but still not enough, we leave it to PCA to eliminate collinearity.

**IMPORTANT**: **DepDelay** is a predictor and **ArrDelay** is a lable; they have very high correlation (0.93), so if we just use DepDelay to predict ArrDelay, the result is likely to have a good score. But in this case, DepDelay just happens few hours before ArrDelay and will not give any significant practical value of prediction (We should be able to predict ArrDelay even days before the departure). So we want to see if other features can be a good predictors to ArrDelay, **delete DepDelay is the choice**.

Để phân loại xem chuyến bay có bị trễ hay không, chúng tôi chọn **ArrTime làm nhãn**.

ActualElapsedTime, CRSElopedTime, AirTime và Khoảng cách có tương quan rất cao (> 0,95) vì vậy chúng tôi chỉ có thể giữ một tính năng (**Khoảng cách) làm công cụ dự đoán**.

CRSDepTime và DepTime cũng có điểm tương quan 0,97, chúng tôi chỉ **giữ DepTime vì đây là dữ liệu thời gian thực.**

CRSArrTime và ArrTime cũng có 0,87 điểm tương quan, chúng tôi chỉ **giữ ArrTime vì đây là dữ liệu thời gian thực.**

ArrTime và DepTime mặc dù có tương quan cao (0,72) nhưng vẫn chưa đủ, chúng tôi để nó cho PCA để loại bỏ cộng tác.

QUAN TRỌNG: DepDelay là một công cụ dự đoán và ArrDelay là một câu chuyện ngụ ngôn; chúng có tương quan rất cao (0,93), vì vậy nếu chúng ta chỉ sử dụng DepDelay để dự đoán ArrDelay, kết quả có thể sẽ có điểm cao. Nhưng trong trường hợp này, DepDelay chỉ xảy ra vài giờ trước ArrDelay và sẽ không đưa ra bất kỳ giá trị thực tế quan trọng nào của dự đoán (Chúng ta có thể dự đoán ArrDelay ngay cả vài ngày trước khi khởi hành). Vì vậy, chúng tôi muốn xem liệu các tính năng khác có thể là một công cụ dự đoán tốt cho ArrDelay hay không, xóa DepDelay là lựa chọn.