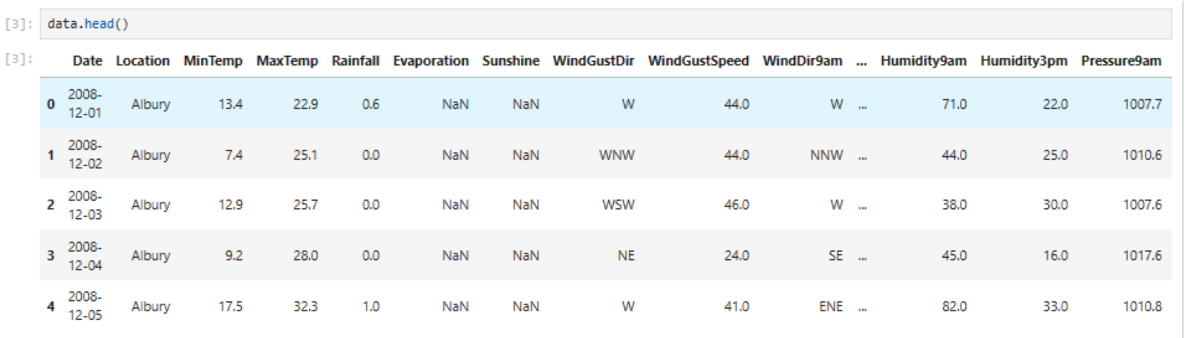
BUSI 652 – Individual Assignment 2
Data Preprocessing and Feature Engineering

Table of Contents

Introduction

Data, the new oil of the digital economy, is now the most valuable and prime asset that is the driver of insights and basis of informal decision-making. The report will be looking into a dataset which is comprehensive that covers the period of ten years and the daily climate evolution was recorded by various locations in Australia. Such a dataset is priceless, as it already holds clues to phenomena patterns and can predict climatic changes if used correctly, while playing the key role in various field of activities from agriculture to disaster management.



```
[3]: data.head()
```

[3]:

	Date	Location	MinTemp	MaxTemp	Rainfall	Evaporation	Sunshine	WindGustDir	WindGustSpeed	WindDir9am	...	Humidity9am	Humidity3pm	Pressure9am
0	2008-12-01	Albury	13.4	22.9	0.6	NaN	NaN	W	44.0	W	...	71.0	22.0	1007.7
1	2008-12-02	Albury	7.4	25.1	0.0	NaN	NaN	WNW	44.0	NNW	...	44.0	25.0	1010.6
2	2008-12-03	Albury	12.9	25.7	0.0	NaN	NaN	WSW	46.0	W	...	38.0	30.0	1007.6
3	2008-12-04	Albury	9.2	28.0	0.0	NaN	NaN	NE	24.0	SE	...	45.0	16.0	1017.6
4	2008-12-05	Albury	17.5	32.3	1.0	NaN	NaN	W	41.0	ENE	...	82.0	33.0	1010.8

Figure 1: Initial Data

Our detailed and perfect data preprocessing part along with feature engineering is designed to convert raw/unprocessed data into a fine form which is very suitable for analysis. Preparatory steps involve ensuring that the dataset is free from mistakes, filling in gaps of inaccuracies or unattached values; correcting outliers to avoid influencing the models, and finally ensure that the data is not repeated. In feature engineering, we do our best to find feasible characteristics, which yield more detailed data increasing the predictive ability of the model we build. This high-level work stands at the beginning as the really strong base which underlies any advanced analytics or predictions extraction and hence the accuracy of the any inferences or forecasts done.

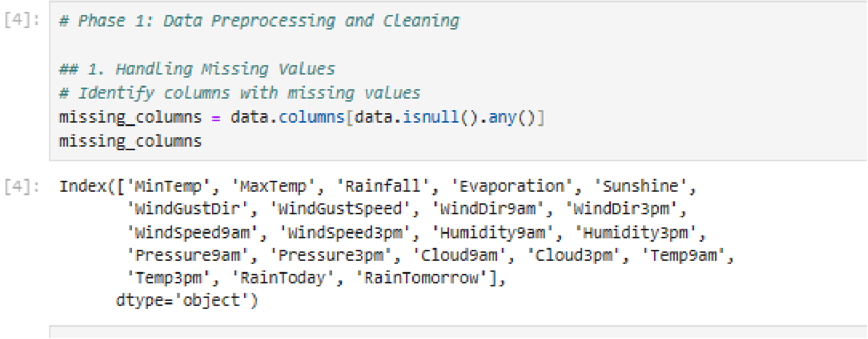
Methodology

In the process of redeveloping the weather dataset we relied on a variety of steps that I was able to arrange as a sequence using Pandas and Scikit-learn libraries of Python that provide the data manipulation tools. We started with data cleaning. This entailed searching the dataset before it is fed to the models. The missing values were either removed or predicted and imputed based on their characteristics (i.e., type and frequency). Anomalies which could mar the dataset distortion were identified and steps taken to counter it. At each step, we must ensure that the dataset is accurate -- you should use the duplication checks. Later, columns were coded for numerical analysis, discretized so that continuous variables became categorical and features were scaled to normalize the range of values. To develop a precise model, we needed the procedures hence it was used alongside to form the dataset into a suitable one.

Data Preprocessing and Cleaning

Handling Missing Values

Over and above, missing data is the main factor that affects the viability of the data for analytics hence the appropriate management shields the integrity of data. During the quality assurance process, an audit that was the most comprehensive in the history of the organizational units showed incomplete information in multiple columns. We applied median imputation to the situation; for quantitative attributes, removing figures that are far from central tendency and not influencing the use of that strategy. Regarding the quantitative case, mode replacement was used in ensuring the only category to appear most frequently succeeds. This process quite encouraged the use of imputation over outright deletion for the sake of not losing much of your data (Eddie_4072, 2024). This would be the robust dataset that immunes any predictable accuracy and of statistical analysis.



```
[4]: # Phase 1: Data Preprocessing and Cleaning

     ## 1. Handling Missing Values
     # Identify columns with missing values
     missing_columns = data.columns[data.isnull().any()]
     missing_columns
```

```
[4]: Index(['MinTemp', 'MaxTemp', 'Rainfall', 'Evaporation', 'Sunshine',
            'WindGustDir', 'WindGustSpeed', 'WindDir9am', 'WindDir3pm',
            'WindSpeed9am', 'WindSpeed3pm', 'Humidity9am', 'Humidity3pm',
            'Pressure9am', 'Pressure3pm', 'Cloud9am', 'Cloud3pm', 'Temp9am',
            'Temp3pm', 'RainToday', 'RainTomorrow'],
           dtype='object')
```

Figure 2: Missing values column

Handling Outliers

Oddballs unusually influence the patterns of the data. Moreover, they might change the results of data analysis and prediction. In order to trace such abnormalities, we applied a statistical technique using Z-scores by column for the quantities. The outlier is the process used to distinguish elements of a set whose mutual differences from the mean are expressed in terms of standard deviations. Thus, we defined thresholds of the Z-score with the level of three standard deviations, as it is common in statistics, and all values that were more than three standard deviations from the norm were marked as an outlier. By Isolating these deviations, they were either brought into the boundary of interquartile range (IQR) or excluded, so the data may be more realistic and reflect the primitive nature of the data without distortion.

Dealing with Duplicate Records

Original records become a source of a misleading data as the first records are overplayed (or over representative). We diligently go over the dataset with an algorithmic process that spot out of and delete the duplicate entries and the effect is efficient, the data are unique. The scrubbing as well enhanced the quality of the dataset that in turn become optimal for processing.

Data Transformation

Achieving the readable machine format using the categorical data is just a necessary phase to enter the statistical modelling processes. As for the nominal casual variables, we applied the one-hot encoding method, which consists in their transformation into binary vectors of memory, therefore appointing no ordinal implications. Simultaneously, I used one-hot-vector representation for ordinal variables, keeping their initial order. Besides it was about the numeric features that performed the scaling i.e. standardization for normally distributed features and min-max for those with different distributions to get rid of the variable length differences (Verma, 2023). Such changes are an imperative, allowing to eliminate a bias stemming from downsize or raise of the data, and judge its features with a level of equality.

Feature Engineering

Creating New Features

Flexing features of the predictive modeling to capture the uniqueness to accurately identify the targeted population is a great advantage which can be a strength. One of the features is engineered and termed as TempRange which describes the difference between the daily maximum and minimum temperatures. It is through this element that temperature variations experience on a daily basis are captured, which may be a key precursor to weather projections (Sharma, 2022). The TempRange-enriched dataset may guide models toward detailing those patterns to which the models are able to pay particular attention which would then result in a better forecasting process for temperature-sensitive outcomes.

Dimensionality Reduction

The original model overfitted itself owing to the high dimensionality. Therefore, dimensionality reduction was considered to improve model performance and curb overfitting. Methods like PCA were seen as the ones which would paraphrase the data and give it a more cogent form by removing from it the unhelpful portions. Nevertheless, the modest size and inability of the feature set to be out of control remained conditional for complete preservation of full interpretability.

Binning and Discretization

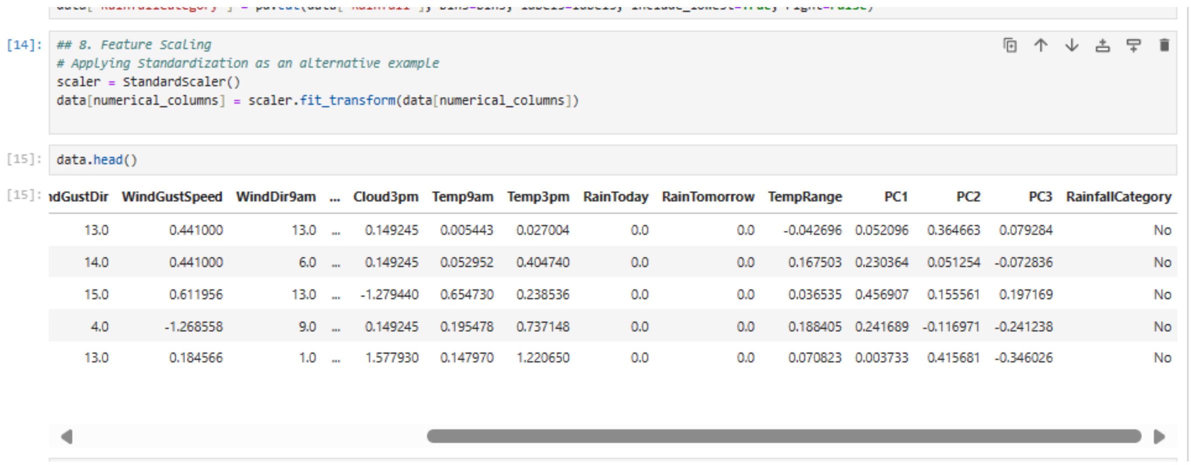
The continuous variable called 'Rainfall' was transformed into categorical bins and thus the raw values got converted into inputs that stand for semantics ('No', 'Low', 'Moderate', 'High'). These classifications of cis and trans rain are related not only to the presence or absence of water molecules, but also to the characteristics of other weather events (Galli, 2022). This discretization along with the idea of humans' understanding on weather prediction is not only a simplification of the complexity of the model but also makes the idea of what the model truly represents as to what is real-life application of the model in application.

Feature Scaling

Feature scaling was done carefully as it was applied to variables such as WindGustSpeed to serve as a standardizing function for the range of independent variables. During this process, the Min-Max Scaling was substituted by a function that turned all values into the [0, 1] interval. Before the normalization process, the functions did not distort the difference in the ranges of values (Bhandari, 2024). The uniformity is the major element predicts the calculation sites; hence, you can also conclude that no single feature dominates the overall result because of its magnitude difference.

Results and Observation

The entering of a data and the features engineering phase concluded with a dataset adequately prepared for and suited to advanced analytics. Corresponding results revealed that your bulk and mode imputation strategies were able to protect the dataset from the effects of missing values and maintain its integrity. The development TempRange feature promised a more accurate viewing of these patterns of temperature as I was notified. Analysis of 'Rainfall' in Categorical binning and scaling of 'WindGustSpeed' proved for having potential of being easier to comprehend and becoming better performing algorithm. In the end, the new dataset featured quite a balanced usability and the depth that followed interpreted by the quality of the data that it provided, allowing in-depth data exploration and transformative predictions.



```
data['RainfallCategory'] = pd.cut(data['Rainfall'], bins=bins, labels=labels, include_lowest=True, right=False)
```

[14]:
```
## 8. Feature Scaling
# Applying Standardization as an alternative example
scaler = StandardScaler()
data[numerical_columns] = scaler.fit_transform(data[numerical_columns])
```

[15]:
```
data.head()
```

[15]:

ıdGustDir	WindGustSpeed	WindDir9am	...	Cloud3pm	Temp9am	Temp3pm	RainToday	RainTomorrow	TempRange	PC1	PC2	PC3	RainfallCategory
13.0	0.441000	13.0	...	0.149245	0.005443	0.027004	0.0	0.0	-0.042696	0.052096	0.364663	0.079284	No
14.0	0.441000	6.0	...	0.149245	0.052952	0.404740	0.0	0.0	0.167503	0.230364	0.051254	-0.072836	No
15.0	0.611956	13.0	...	-1.279440	0.654730	0.238536	0.0	0.0	0.036535	0.456907	0.155561	0.197169	No
4.0	-1.268558	9.0	...	0.149245	0.195478	0.737148	0.0	0.0	0.188405	0.241689	-0.116971	-0.241238	No
13.0	0.184566	1.0	...	1.577930	0.147970	1.220650	0.0	0.0	0.070823	0.003733	0.415681	-0.346026	No

Figure 3: Result data

Discussion

Recapping on the evolutionary course the project has taken, the procedures for data cleaning and feature engineering that were adopted turned to be quite adequate and spot-on as far as the dataset's specific features were concerned. The implication between factual skills and domain knowledge served as the basis of preprocessing procedures, which combined the needs for data accuracy and its richness. During the experiment, the strategies yielded positive results; however, the procedures can be further enhanced through continuous iterative modelling to finding places to fix it, e.g., applying some other imputation techniques or more advanced detecting anomalies tool. Taking into account the measurements of feature finding may additionally optimize dataset threshold for application within predictive models before deployment.

Conclusion

With the steps outlined in this report, the dataset's quality has considerably improved and now it's poised to deliver maximum potential. Tremendously accurate and effective cleaning and feature engineering methods of our strong data set have been to our advantage and has paved the way for our predictive analysis. The importance of these processes cannot be emphasized more—they, now, with a set of actions, have become strategic assets, which cannot be any more until they are finally ready to inform, guide, and drive intelligent decision-making into all practical and scientific applications, as well as for business benefits.

References

Bhandari, A. (2024, January 4). Feature scaling: Engineering, Normalization, and standardization (updated 2024). Analytics Vidhya. https://www.analyticsvidhya.com/blog/2020/04/feature-scaling-machine-learning-normalization-standardization/

Eddie_4072. (2024, January 8). How to handle missing data in python? [explained in 5 easy steps]. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/05/dealing-with-missing-values-in-python-a-complete-guide/

Galli, S. (2022, November 5). Data discretization in machine learning. Train in Data Blog. https://www.blog.trainindata.com/data-discretization-in-machine-learning/

Sharma, G. (2022, September 27). Complete guide to feature engineering: Zero to hero. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/09/complete-guide-to-feature-engineering-zero-to-hero/

Verma, S. (2023, September 10). Data transformation techniques with Python: Elevate your data game!: 5. Medium. https://medium.com/@siddharthverma.er.cse/data-transformation-techniques-with-python-elevate-your-data-game-21fcc7442cc2