Data Visualization Learning Report

In this big data course, I have deeply explored the field of data visualization, and the following is a detailed report on the learning process, task completion, and learning experience from the perspective of students.

I. Learning process and task completion

（一）Microsoft Power BI 实践练习

1. Learning style

- I first followed the learning path (https://learn.microsoft.com/en-us/training/paths/build-power-bi-visuals-reports/) provided by the teacher and carefully studied the basic operations and functions of Power BI. Along the way, I watched every step of the tutorial, including how to connect to data sources, create reports, and use the various visualization types.

- For connecting data sources, I tried connecting to a local Excel file that contains some simulated sales data. During the connection process, I double-checked the file path and data format to make sure that the data was successfully imported.

2. Practical operation and harvest

- When creating a report, I was able to create a variety of visualizations by dragging and dropping different fields into the appropriate areas. For example, I dragged the Product Category field to the Axis area and the Sales field to the Value area to quickly generate a bar chart that clearly shows how sales compare for different product categories. This gave me a visual insight into which product categories were selling well and which needed improvement.

- I also explored Power BI's interactive features, such as adding slicers to dynamically see how sales change over time by selecting different time intervals. This interactivity allows me to analyze the data more deeply and uncover trends and patterns hidden in the data.

(2) Explore the basics of data visualization

1. Use of learning resources

- I carefully studied the link (https://learn.microsoft.com/enus/training/modules/explore-fundamentals-data-visualization/) provided by the teacher on the basics of data visualization. In the process, I learned the basic principles of data visualization, such as the importance of conciseness, accuracy, and intuitiveness.

- I made detailed notes for the applicable scenarios for different chart types. For example, I learned that line charts are good for showing trends in time series data, and pie charts are good for showing the proportions of each part to the whole.

2. Guidance on practice

- These basics became an important guide for me in subsequent visualization tasks. When I'm faced with a set of data, I can quickly choose the right type of visualization based on the characteristics of the data and the information I want to express. For example, when analyzing the distribution of students' grades, I chose a bar chart to show the comparison of grades in each subject, and a box chart to show the dispersion of grades according to the distribution of grades in different subjects.

(3) Data processing and visualization with Anaconda Spyder

1. Installation and Launch

- I followed the installation guide (https://docs.spyder-ide.org/current/installation.html) to install Anaconda Spyder on my laptop. During the installation process, I encountered some problems with dependencies, but by carefully reading the error message, I searched the Internet for relevant solutions, and finally successfully solved the problem and launched Spyder without any problems.

- After launching, I spent some time familiarizing myself with the layout of Spyder's interface, including the location and purpose of functional areas such as the code editing area, variable browser, console, etc.

2. Dataset import and exploration

- Downloaded the Stroke dataset from the GitHub repository and placed it in the same working directory as the Python file. Then use the 'pd.read_csv' function to successfully import the dataset, and see in the Variable Browser that the data type is 'DataFrame' and contains multiple fields. By double-clicking on the variables, I looked at some of the data in the dataset in detail and gained a preliminary understanding of the content and structure of the data.

3. Data visualization operations

- **Show the first five rows of records in the dataset**: Using the 'print(data.head(5))' command, I can clearly see the first five rows of data in the dataset, which gives me a more intuitive understanding of the format and content of the data, such as the gender, age, and whether the patient has high blood pressure or not, and how information is presented in the dataset.

- **Create a histogram**

  - Run the data.hist() command to create a histogram of all numeric fields, and observe the distribution of data on each numeric

field. However, I found that the default histogram is not easy to read, for example, the histogram distribution of the 'id' field is scattered.

- For the 'avg_glucose_level' field, I recreated the histogram by adding the 'bins = 20' parameter, so that I could see the distribution of the data more clearly, and found that the average blood glucose levels of most patients were concentrated in a certain range.

- To compare the distribution of multiple variables, I created a histogram with 'avg_glucose_level' and 'bmi'. I first created a new data frame 'data1 = data[['avg_glucose_level', 'bmi']]'', and then used the 'data1.plot.hist()' command to successfully plot a histogram comparing the distributions of the two variables, which helped me analyze the possible relationship between the two variables.

- **Create a Histogram**

- For the categorical data 'Residence_type', I first used 'print(data["Residence_type"].value_counts())' to count the number of different dwelling types, and then used 'data["Residence_type"].value_counts().plot(kind="bar")' to plot a histogram, A visual comparison of the difference in the number of patients between urban and rural residential types was performed.

- 在使用`matplotlib.pyplot`库创建工作类型柱状图时，我按照教程中的代码`ax = data['work_type'].value_counts().plot(kind='bar', title="Number of Work Types")`进行操作，并设置了轴标签

（`ax.set_xlabel("Work Type")`和`ax.set_ylabel("Frequency")`），使图表更加规范、易读。 我还尝试了其他创建柱状图的方式，如`data.groupby(['work_type', 'work_type'])['work_type'].count().unstack(0).plot.bar`，进一步理解了如何对数据进行分组统计并可视化。

- Create a scatter plot: A scatter plot of age and body mass index was created using data.plot.scatter(x="age", y="BMI"), and no obvious strong correlation was found from the plot, and the distribution of data points was scattered. I then tried to compare other numerical variables, such as 'hypertension' and 'avg_glucose_level', and by creating a scatterplot analysis, although I initially found that there seemed to be a certain positive correlation between the two, I also realized that further data validation and analysis were needed to determine the reliability of this relationship.
- **Create a pie chart**
  - Taking the work type as an example, we used 'data["work_type"].value_counts().plot(kind="pie")' to create a pie chart to show the proportion of different work types in the dataset.
  - In order to optimize the display of the pie chart, I set parameters such as 'ax.xaxis.set_visible (False)', 'ax.yaxis.set_visible (False)', and 'autopct='%1.0f%%'' according to the tutorial, so that the

pie chart is more concise and beautiful, and the information is clear, and I can clearly see the specific proportion of each work type.

(4) Spotify Dataset Visualization Practice (Optional)

1. Dataset import and preliminary exploration

- I downloaded the Spotify dataset and made sure the dataset and the Python file were in the same working directory. Successfully import data using 'import pandas as pd' and 'data = pd.read_csv('spotify.csv")'. Looking at the type and content of the data in the Variable Browser, I noticed that the dataset contained information such as the number of plays of the songs, the release date, and some 'nan' values, indicating that the data was missing.

2. Visual operation and analysis

- When it comes to drawing line charts, I try a variety of ways. First of all, I used 'data.plot(x="Date", kind="line", figsize=(14,6))' to plot the trend of all numeric columns over time, which allowed me to see how the overall data changed in the time dimension.

- I then plotted the playback of specific songs (e.g. 'Shape of You' and 'Despacito') over time, by specifying parameters such as 'y=["Shape of You", "Despacito"]', and added appropriate axis labels ('xlabel' and 'ylabel') and title ('title') so that the playback of different

songs could be compared more clearly, Changes in their popularity at different time points were observed.

（五）完成金融数据分析工作坊（Financial Data Analysis with Spyder）

1. Learning process and challenges

- I followed the requirements of the workshop (https://docs.spyder-ide.org/current/workshops/financial.html) to learn the relevant knowledge of financial data analysis step by step. In the process, I encountered some challenges related to the understanding and processing of financial data, such as how to correctly interpret financial indicators, how to deal with seasonal factors in time series data, etc.

- However, I gradually overcame these difficulties by consulting relevant financial knowledge materials and combining them with the instructions for using financial data analysis libraries in Python (such as 'pandas_finance', etc.).

2. Practical results and gains

- I successfully used Spyder to analyze some simulated financial data, such as calculating the return on stocks, plotting the risk-return curve of my portfolio, etc. Through these practical operations, I not only improved my data processing and analysis skills in the financial

field, but also gained a deeper understanding of the operation mechanism of the financial market, which accumulated valuable experience for my future study or work in related fields.

（六）Anaconda Spyder 与 Microsoft Power BI 的比较

1. Comparison of functional features

- Anaconda Spyder: Based on the Python language, this makes it extremely flexible and extensible. I can take advantage of Python's rich library ecosystem, such as 'numpy' for numerical computation, 'matplotlib' and 'seaborn' for advanced visualization, etc., to achieve a variety of complex data processing and customized visualization needs. Spyder's advantages are especially evident when dealing with data in special formats, such as geospatial data, or when complex data mining algorithms need to be implemented. The interactive development environment allows me to easily code, debug, and explore data, and see the values of variables and how the data changes in real time, which is very helpful for in-depth analysis of data.

- Microsoft Power BI: A professional business intelligence tool that offers a wide variety of visualization types and powerful report creation capabilities. It's relatively simple, and intuitive drag-and-drop allows you to quickly create beautiful, professional visualizations. It

supports the connection of multiple data sources, which is very convenient for enterprises to integrate data from different systems. For example, companies can easily integrate sales data from databases, market research data in Excel files, etc., together for analysis and visualization. Its good collaboration and sharing capabilities also make it easy for team members to edit and view reports together, facilitating information flow and decision-making within the enterprise.

2. Differences in applicable scenarios

- Anaconda Spyder: Ideal for the pre-exploratory phase of academic research and data analysis projects. When I need to conduct in-depth analysis of the data, try different data processing methods and algorithms, and explore the internal relationships and characteristics of the data, Spyder can provide great support. For example, when working on a machine learning project, I can use Spyder for data preprocessing, feature engineering, and preliminary data analysis to inform subsequent model building. Spyder is an excellent choice for handling tasks that require a high degree of customization, such as data processing in specific formats or complex data mining algorithm implementations.

- Microsoft Power BI: More suitable for enterprise-level business intelligence application scenarios, such as daily operation

management, decision support, etc. Enterprise managers can use Power BI to quickly create interactive dashboards and reports, monitor key business metrics (such as sales, profits, market share, etc.) in real time, and identify problems and make decisions in a timely manner. When you need to quickly create visual reports and share them with team members or external partners, the benefits of Power BI are clear. For example, in project debriefing, using Power BI to display information such as project progress, resource allocation, and risk assessment can make debriefing more intuitive and vivid, and improve communication efficiency.

3. Learning difficulty and resources

- Anaconda Spyder: It is relatively difficult to learn and requires some basic knowledge of Python programming. During my learning process, I spent time learning the basic syntax of the Python language, data structures, and how to use the associated data processing and visualization libraries. However, Python, as a widely used programming language, has a wealth of learning resources to refer to, such as online tutorials, books, forums, etc. For example, I have deepened my understanding of Python data analysis by reading books such as "Python Data Analysis Basics", so that I can better process and visualize data in Spyder. The documentation (https://docs.spyder-ide.org/current/index.html) of Spyder itself is

also very detailed, with lots of instructions and tutorials to help me get started quickly.

- Microsoft Power BI: Relatively low learning difficulty, user-friendly interface, simple and intuitive operation. Microsoft provides a large number of official documentation (such as https://learn.microsoft.com/en-us/training/paths/build-power-bi-visuals-reports/) and online training resources through which I can quickly learn and master its features. For example, when I was learning Power BI, I took some online training courses that systematically explained the knowledge of report creation, data modeling, and visual design in Power BI, so that I could master the basic operations and create valuable visualization reports in a relatively short period of time.

Second, learning and experience

(1) Data visualization skills improvement

1. Be proficient in using Microsoft Power BI and Anaconda Spyder for data visualization operations, and select appropriate tools and visualization types according to data characteristics and analysis needs. For example, when working with large-scale business data, I

prioritize using the power of Power BI to quickly create intuitive reports; Spyder's flexibility allows me to dig deeper into my data.
2. Master a variety of visualization chart creation and optimization methods, such as histograms, bar charts, scatter charts, pie charts, etc. I learned how to adjust the parameters of the chart to make it more accurate and clear. For example, when creating a histogram, the division of data intervals is optimized by adjusting the 'bins' parameter to make the data distribution more intuitive. In pie charts, set labels and percentage displays to enhance the readability of the charts.

(2) Cultivation of data analysis thinking

1. In the process of data visualization, I gradually cultivated a data analysis mindset. By looking at the visualizations, I was able to spot patterns, trends, and outliers in the data to make sound assumptions and questions, and to further analyze the data. For example, when analyzing a visualization of sales data, I found that the sales of a product suddenly dropped during a specific time period, which prompted me to further explore the cause, which could be increased market competition, product quality issues, or marketing strategy adjustments, etc., so as to provide a valuable reference for decision-making.

2. Learn how to analyze data from different angles, and explore the potential information behind the data by comparing the relationships between different variables. For example, when analyzing patient data, try to find out the relevant factors that affect the occurrence of diseases by comparing the relationship between variables such as age, gender, and disease indicators, so as to provide data support for medical research.

(3) Enhance problem-solving skills

1. In the process of learning and practicing, I encountered various problems, such as software installation problems, code errors, data understanding difficulties, etc. By constantly reviewing materials, searching for solutions, debugging code, and consulting teachers and classmates, I gradually improved my problem-solving skills. For example, when I had a dependency conflict issue while installing Anaconda Spyder, I successfully resolved the problem by carefully reading the error prompts, searching the web for relevant solutions, and trying different workarounds.
2. This problem-solving ability is not only very important in this course, but will also have a positive impact on my future studies and work. When faced with complex data analysis tasks and technical

challenges, I am confident that I can find solutions through my own efforts and continuously improve my capabilities.

(4) A deep understanding of the importance of data visualization

1. Data visualization is a powerful way to turn complex data into intuitive information, which can help people in different fields better understand data and make more informed decisions. In the business field, visualization can assist business managers to analyze market trends and optimize product strategies. In the medical field, it helps doctors diagnose diseases and evaluate the effect of treatment; In the field of scientific research, it can promote the presentation and exchange of research results.
2. Through this study, I deeply realized that data visualization is not only about creating beautiful charts, but more importantly, it is more important to accurately convey data information and guide the audience to correctly understand the meaning behind the data. Therefore, in the future visualization practice, I will pay more attention to following the basic principles of visualization design to ensure the effectiveness and reliability of data visualization.

3. Summary and outlook

Through this course, I have made significant progress in the field of data visualization. Not only did I master the use of a variety of data visualization tools, but I also developed data analysis thinking and problem-solving skills, and I deeply understood the importance of data visualization. In my future studies and work, I will continue to learn more about data visualization, explore new visualization techniques and methods, and improve my data visualization skills. At the same time, I will also apply data visualization skills to practical projects, provide strong support for solving practical problems, and contribute to the development of related fields.