ChatGPT Training Tasks for Data Professionals (1 Hour Workshop)

Beginner Level (15 minutes)

Task 1: Data Cleaning Assistant (5 minutes)

- Given this messy dataset:

```
date,sales_amt,region
2024-01-01,$1,200.00,North
2024-01-02,1500.50,north
2024/01/03,1,800.75,NORTH
null,2000.25,North
```

- Ask ChatGPT to:
  a) Identify data quality issues
  b) Generate Python code to clean the data
  c) Suggest data validation rules
- Learning Goal: Understanding how to use ChatGPT for data cleaning tasks

Task 2: SQL Query Optimization (5 minutes)

- Present this basic SQL query:

```sql
SELECT
  customers.*,
  orders.order_date,
  orders.amount
FROM customers
LEFT JOIN orders ON customers.id = orders.customer_id
WHERE orders.amount > 1000
ORDER BY orders.order_date
```

- Ask ChatGPT to:
  a) Optimize the query

b) Add appropriate indexing suggestions
c) Explain the improvements

- Learning Goal: Learning to use ChatGPT for query optimization

Task 3: Data Pipeline Debug (5 minutes)

- Share a simple ETL error scenario:

```python
def transform_data(df):
    df['date'] = pd.to_datetime(df['date'])
    df['revenue'] = df['quantity'] * df['price']
    df.groupby('category')['revenue'].sum()
    return df
```

- Ask ChatGPT to:
  a) Identify potential issues
  b) Suggest error handling
  c) Improve the code
- Learning Goal: Using ChatGPT for code review and debugging

Intermediate Level (25 minutes)

Task 4: Feature Engineering (8 minutes)

- Present a dataset for customer churn prediction:

```
columns: [registration_date, last_purchase_date, total_purchases,
average_order_value, support_tickets]
```

- Ask ChatGPT to:
  a) Suggest relevant features
  b) Generate feature engineering code
  c) Explain the significance of each feature
  d) Recommend feature scaling approaches
- Learning Goal: Leveraging ChatGPT for feature engineering ideas

Task 5: Data Architecture Design (8 minutes)

- Present a scenario: "Design a data warehouse for an e-commerce platform"
- Ask ChatGPT to:
  a) Suggest a dimensional model
  b) Create table definitions
  c) Define slowly changing dimensions strategy
  d) Recommend partitioning strategy
- Learning Goal: Using ChatGPT for architecture planning

Task 6: Performance Monitoring (9 minutes)

- Share a monitoring scenario:

```python
# Current monitoring code
def monitor_pipeline():
    log.info("Pipeline started")
    process_data()
    log.info("Pipeline completed")
```

- Ask ChatGPT to:
  a) Add comprehensive metrics collection
  b) Implement error tracking
  c) Create alerting logic
  d) Generate dashboard queries
- Learning Goal: Enhancing monitoring systems with ChatGPT

Advanced Level (20 minutes)

Task 7: Machine Learning Pipeline (7 minutes)

- Present a classification problem with imbalanced data
- Ask ChatGPT to:
  a) Generate preprocessing steps
  b) Suggest model selection strategy
  c) Create cross-validation code
  d) Implement evaluation metrics
- Learning Goal: Building ML pipelines with ChatGPT assistance

Task 8: Data Quality Framework (5 minutes)

- Present requirements for automated data quality checks
- Ask ChatGPT to:
  a) Design test cases
  b) Generate testing code
  c) Create reporting framework
  d) Suggest implementation strategy
- Learning Goal: Developing quality frameworks with ChatGPT

Task 9: Real-time Analytics (5 minutes)

- Present a streaming data scenario
- Ask ChatGPT to:
  a) Suggest architecture components
  b) Create data processing logic
  c) Design aggregation strategies
  d) Implement windowing functions
- Learning Goal: Understanding streaming analytics design with ChatGPT

Task 10: Integration Challenge (3 minutes)

- Combine multiple aspects into a complete solution:
  "Design an end-to-end pipeline for processing customer transaction data"
- Ask ChatGPT to:
  a) Create architecture diagram code
  b) Generate key component implementations
  c) Define monitoring strategy
  d) Suggest deployment approach
- Learning Goal: Building complete solutions with ChatGPT

Assessment Criteria:

- Code quality and completeness
- Architecture design principles
- Performance considerations
- Scalability approaches
- Error handling and resilience

Tips for Data Professionals:

1. Always verify generated code
2. Use ChatGPT for brainstorming architectural approaches
3. Ask for explanations of suggested solutions
4. Iterate on responses to improve solutions
5. Combine ChatGPT suggestions with best practices
6. Test generated code with small datasets first

Learning Objectives:

- Efficient data pipeline development
- Code optimization and debugging
- Architecture design patterns
- Best practices implementation
- Problem-solving approaches

Note: Participants should have basic understanding of Python, SQL, and data concepts.