2-Day Data Science Interview Preparation Guide

DAY 1 – CORE FOUNDATIONS & EDA

Morning (3 hrs) – Math & Stats Refresher

Content to Cover:

- **Probability & Distributions**
  - Basic probability rules (addition, multiplication, conditional)
  - Normal, binomial, Poisson distributions
  - Central Limit Theorem
- **Descriptive Statistics**
  - Mean, median, mode, variance, standard deviation
  - Quartiles, percentiles, IQR
- **Inferential Statistics**
  - Hypothesis testing (null/alternative hypotheses)
  - Type I & Type II errors
  - P-values and statistical significance
  - Confidence intervals (95%, 99%)
  - t-tests, chi-square tests

Resources:

1. **Khan Academy Statistics & Probability** (Free)
   - Focus on: Probability, Random Variables, Sampling Distributions

- URL: khanacademy.org/math/statistics-probability

2. **StatQuest YouTube Channel** (Free)
   - Watch: "Hypothesis Testing", "P-values", "Confidence Intervals"
   - Creator: Josh Starmer - excellent visual explanations
3. **Think Stats (Free PDF)**
   - By Allen Downey - practical statistics for programmers
   - Download from: greenteapress.com/thinkstats/

Practice Problems (15 problems):

1. **Basic Probability (5 problems)**
   - Calculate probability of drawing cards, rolling dice
   - Conditional probability scenarios
   - Use: Practice problems from Khan Academy
2. **Distribution Problems (5 problems)**
   - Normal distribution z-score calculations
   - Binomial probability calculations
   - Use: StatTrek.com probability calculator for verification
3. **Hypothesis Testing (5 problems)**
   - One-sample t-tests
   - Two-sample comparisons
   - Use: OpenIntro Statistics textbook (free PDF) practice problems

How to Practice:

- **Time allocation:** 1.5 hrs theory + 1.5 hrs problems

- **Method:**
  - Watch 2-3 StatQuest videos (30 min)
  - Read key concepts from Think Stats (60 min)
  - Solve practice problems with pen/paper (60 min)
- **Self-check:** Explain each concept aloud in simple terms

Afternoon (3 hrs) – Exploratory Data Analysis (EDA)

Content to Cover:

- **Pandas Operations**
  - Data loading, inspection (head, info, describe)
  - Filtering with boolean indexing
  - GroupBy operations and aggregations
  - Merge, join, concat operations
  - Pivot tables and cross-tabulations
- **Data Cleaning**
  - Identifying missing data patterns
  - Handling missing values (drop, fill, interpolate)
  - Outlier detection (IQR method, Z-score)
  - Data type conversions
- **Feature Engineering Basics**
  - Creating new features from existing ones
  - Binning/bucketing continuous variables

- Encoding categorical variables
- Date/time feature extraction

Resources:

1. **Python Data Science Handbook** (Free online)
   - Chapter 3: Data Manipulation with Pandas
   - URL: jakevdp.github.io/PythonDataScienceHandbook/
2. **Kaggle Learn - Pandas Course** (Free + Certificate)
   - Interactive exercises with real datasets
   - URL: kaggle.com/learn/pandas
3. **Real Python Pandas Tutorials**
   - "Working with Missing Data in Pandas"
   - "Pandas GroupBy: Your Guide to Grouping and Splitting Data"

Hands-on Practice Datasets:

Dataset 1: Titanic (1.5 hrs)

- **Source:** Kaggle Titanic Competition
- **Tasks:**

```python
```

```python
# 1. Basic exploration
df.info(), df.describe(), df.head()

# 2. Missing data analysis
df.isnull().sum()
sns.heatmap(df.isnull())

# 3. Survival analysis by different features
df.groupby('Sex')['Survived'].mean()
df.groupby('Pclass')['Survived'].mean()

# 4. Feature engineering
df['FamilySize'] = df['SibSp'] + df['Parch'] + 1
df['Title'] = df['Name'].str.extract('([A-Za-z]+)\.')

# 5. Visualizations
import seaborn as sns
sns.countplot(x='Survived', hue='Sex', data=df)
sns.boxplot(x='Survived', y='Age', data=df)
```

Dataset 2: Boston Housing (1.5 hrs)

- **Source:** sklearn.datasets or Kaggle
- **Tasks:**

```python
```

```
# 1. Correlation analysis
correlation_matrix = df.corr()
sns.heatmap(correlation_matrix, annot=True)

# 2. Outlier detection
Q1 = df.quantile(0.25)
Q3 = df.quantile(0.75)
IQR = Q3 - Q1
outliers = df[((df < (Q1 - 1.5 * IQR)) | (df > (Q3 + 1.5 * IQR))).any(axis=1)]

# 3. Feature relationships
sns.pairplot(df)
sns.scatterplot(x='RM', y='PRICE', data=df)
```

Practice Environment Setup:

1. **Google Colab** (Recommended for beginners)
   - No setup required, GPU/TPU available
   - Pre-installed libraries
   - Easy sharing and collaboration
2. **Jupyter Notebook locally**

```bash
pip install pandas numpy matplotlib seaborn jupyter
jupyter notebook
```

3. **Kaggle Notebooks**
   - Access to competition datasets

- Community notebooks for reference
- Built-in data visualization tools

Evening (2 hrs) – ML Concepts (Theory)

Content to Cover:

Learning Types (30 min)

- Supervised Learning (classification, regression)
- Unsupervised Learning (clustering, dimensionality reduction)
- Reinforcement Learning (basic concept)
- Semi-supervised and self-supervised learning

Core Algorithms (60 min)

- **Linear Regression**
  - Assumptions, interpretation of coefficients
  - R-squared, residual analysis
- **Logistic Regression**
  - Sigmoid function, odds ratio
  - Maximum likelihood estimation
- **Decision Trees**
  - Splitting criteria (Gini, entropy)
  - Pruning, interpretability
- **Random Forest**

- Bagging, feature importance
- Bias-variance tradeoff

Key Concepts (30 min)

- Overfitting vs Underfitting
- Bias-Variance Tradeoff
- Regularization (L1/L2)
- Cross-validation

Resources:

1. **Andrew Ng's Machine Learning Course** (Coursera)
   - Week 1-3 videos (focus on linear/logistic regression)
   - Mathematical intuition with practical examples
2. **StatQuest Machine Learning Playlist**
   - "Linear Regression", "Logistic Regression", "Decision Trees"
   - "Random Forests", "Cross Validation"
3. **Hands-On Machine Learning (Book)**
   - Chapters 1-4: ML landscape, end-to-end project, classification
   - Available on O'Reilly Learning Platform
4. **Towards Data Science (Medium)**
   - Search for "Machine Learning Fundamentals"
   - Read 3-4 well-rated articles on core algorithms

Study Method:

Time Allocation:
- 30 min: Watch StatQuest videos on linear/logistic regression
- 45 min: Read chapter summaries from textbook/articles
- 30 min: Create concept maps linking algorithms to use cases
- 15 min: Practice explaining concepts aloud (record yourself)

Key Questions to Master:

1. "Explain the difference between classification and regression"
2. "What is overfitting and how do you prevent it?"
3. "When would you use Random Forest vs Logistic Regression?"
4. "Explain bias-variance tradeoff with examples"

DAY 2 – ML DEEP DIVE & TOOLS

Morning (3 hrs) – ML Practice + Evaluation Metrics

Evaluation Metrics to Master:

Classification Metrics (90 min)

- **Confusion Matrix**: TP, TN, FP, FN
- **Accuracy**: When to use and limitations
- **Precision & Recall**: Trade-offs, use cases
- **F1 Score**: Harmonic mean interpretation
- **ROC-AUC**: Curve interpretation, thresholds
- **Specificity, Sensitivity**

Regression Metrics (30 min)

- **RMSE vs MAE**: When to use each
- **R-squared**: Interpretation and limitations
- **Adjusted R-squared**: Multiple features

Cross-Validation (30 min)

- K-fold cross-validation
- Stratified cross-validation
- Time series cross-validation
- Leave-one-out cross-validation

Resources:

1. **Scikit-learn Documentation**
   - Model evaluation guide
   - URL: scikit-learn.org/stable/modules/model_evaluation.html
2. **Google's Machine Learning Crash Course**
   - Classification and Regression modules
   - URL: developers.google.com/machine-learning/crash-course
3. **Neptune.ai Blog**
   - "25 Evaluation Metrics for Binary Classification"
   - Comprehensive guide with code examples

Hands-on End-to-End ML Project (2 hrs)

Project: Predicting Customer Churn

python

```
# Complete workflow to practice:

# 1. Data Loading & EDA (20 min)
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, roc_auc_score

# Load dataset (use Telco Customer Churn from Kaggle)
df = pd.read_csv('telco_churn.csv')

# 2. Data Preprocessing (30 min)
# Handle missing values
# Encode categorical variables
# Feature scaling if needed
from sklearn.preprocessing import LabelEncoder, StandardScaler

# 3. Model Training (30 min)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Try multiple models
models = {
    'Logistic Regression': LogisticRegression(),
    'Random Forest': RandomForestClassifier(),
}

# 4. Model Evaluation (30 min)
from sklearn.metrics import confusion_matrix, roc_curve
import matplotlib.pyplot as plt

for name, model in models.items():
```

```
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)

    print(f"{name} Results:")
    print(classification_report(y_test, y_pred))
    print(f"ROC-AUC: {roc_auc_score(y_test, y_pred)}")

    # Plot ROC curve
    fpr, tpr, thresholds = roc_curve(y_test, model.predict_proba(X_test)[:, 1])
    plt.plot(fpr, tpr, label=f'{name} (AUC = {roc_auc_score(y_test, model.predict_proba(X_test)[:, 1]):.3f})')

plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('ROC Curves')
plt.legend()
plt.show()

# 5. Feature Importance & Interpretation (10 min)
feature_importance = pd.DataFrame({
    'feature': X.columns,
    'importance': model.feature_importances_
}).sort_values('importance', ascending=False)

print(feature_importance.head(10))
```

Practice Datasets:

1. **Titanic (Classification)**: kaggle.com/c/titanic
2. **Boston Housing (Regression)**: sklearn.datasets
3. **Heart Disease (Classification)**: kaggle.com/datasets/johnsmith88/heart-disease-dataset
4. **Customer Churn**: kaggle.com/datasets/blastchar/telco-customer-churn

---

Afternoon (3 hrs) – Big Data & Cloud Tools

Big Data Concepts (90 min)

Hadoop Ecosystem

- **HDFS**: Distributed file storage
- **MapReduce**: Distributed computing paradigm
- **Hive**: SQL-like queries on big data
- **Basic architecture and use cases**

Apache Spark

- **RDD, DataFrames, Datasets**
- **Spark SQL, MLlib**
- **Lazy evaluation concept**
- **When to use Spark vs traditional tools**

Distributed Computing Principles

- **Horizontal vs Vertical scaling**
- **Data partitioning and parallelization**
- **CAP theorem basics**

Cloud Platforms Overview (90 min)

AWS Services

- **S3**: Object storage for data lakes

- **EC2**: Compute instances for ML workloads
- **SageMaker**: End-to-end ML platform
- **Redshift**: Data warehousing
- **EMR**: Managed big data platform

Google Cloud Platform

- **BigQuery**: Serverless data warehouse
- **Cloud ML Engine**: Managed ML platform
- **Cloud Storage**: Object storage
- **Dataflow**: Stream/batch data processing

Microsoft Azure

- **Azure ML Studio**: Drag-and-drop ML
- **Data Factory**: Data integration service
- **Synapse Analytics**: Data warehouse
- **Blob Storage**: Object storage

Resources:

1. **Coursera - Big Data Specialization (University of California San Diego)**
   - Week 1 of "Introduction to Big Data" (Free audit)
   - Focus on Hadoop and Spark concepts
2. **AWS Machine Learning Training**
   - "AWS Machine Learning Foundations" (Free)
   - URL: aws.amazon.com/training/learning-paths/machine-learning/

3. **Google Cloud Skills Boost**
    - "BigQuery Basics for Data Analysts" (Free tier available)
    - Hands-on labs with real GCP environment
4. **YouTube Channels:**
    - **Krish Naik**: "Complete Big Data Tutorial"
    - **edureka!**: "Apache Spark Tutorial"
    - **AWS Online Tech Talks**: Cloud ML services overview

Hands-on Practice:

Option 1: Local Spark Setup (Advanced)

```bash
# Install PySpark
pip install pyspark

# Basic PySpark script to understand concepts
from pyspark.sql import SparkSession

spark = SparkSession.builder.appName("DataSciencePrep").getOrCreate()
df = spark.read.csv("titanic.csv", header=True, inferSchema=True)
df.groupBy("Sex").avg("Age").show()
```

Option 2: Cloud Platform Free Tiers

- **AWS Free Tier**: 12 months free, practice with SageMaker
- **Google Cloud**: $300 credit, practice BigQuery
- **Azure**: $200 credit, try Azure ML Studio

Option 3: Databricks Community Edition (Recommended)

- **Free Spark cluster**
- **Pre-loaded datasets**
- **Collaborative notebooks**
- **URL**: community.cloud.databricks.com

Quick Demo Tasks (30 min each):

1. **BigQuery**: Load public dataset, run SQL queries
2. **AWS S3**: Upload dataset, explore through console
3. **Databricks**: Create cluster, run basic Spark operations

---

Evening (2 hrs) – Mock Interview & Revision

Top 20 ML Interview Questions (90 min)

Statistics & Probability (5 questions)

1. "Explain the Central Limit Theorem and its importance"
2. "What's the difference between Type I and Type II errors?"
3. "How do you interpret a p-value?"
4. "When would you use a t-test vs z-test?"
5. "Explain confidence intervals in simple terms"

Machine Learning Fundamentals (8 questions) 6. "What's the bias-variance tradeoff?" 7. "How do you handle overfitting?" 8. "Explain the difference between bagging and boosting" 9. "When would you use Random Forest vs SVM?" 10. "What's the curse of dimensionality?" 11. "How do

you handle imbalanced datasets?" 12. "Explain cross-validation and its types" 13. "What are the assumptions of linear regression?"

Practical ML (7 questions) 14. "How do you evaluate a classification model?" 15. "What's the difference between precision and recall?" 16. "How do you choose the right evaluation metric?" 17. "Explain feature selection techniques" 18. "How do you handle missing data?" 19. "What's the difference between L1 and L2 regularization?" 20. "How do you approach a new ML problem?"

Resources for Answers:

1. **"Cracking the Data Science Interview" (Book)**
   - 500+ practice questions with detailed answers
2. **InterviewBit Data Science Questions**
   - URL: interviewbit.com/data-science-interview-questions/
   - Categorized by difficulty and topic
3. **Towards Data Science - Interview Prep Articles**
   - Search: "Data Science Interview Questions 2024"
   - Multiple comprehensive guides
4. **Glassdoor & LeetCode**
   - Company-specific interview experiences
   - Real questions from Google, Amazon, Facebook

Practice Method (30 min):

The STAR Method for Technical Questions:

- **S**ituation: Set up the problem context

- **T**ask: What needs to be solved
- **A**ction: Your approach and methodology
- **R**esult: Outcome and lessons learned

Example Practice:

> Question: "How would you handle missing data in a dataset?"
>
> Your Answer Structure:
> 1. Identify the type and pattern of missing data (MCAR, MAR, MNAR)
> 2. Quantify the extent (percentage missing per feature)
> 3. Choose appropriate strategy:
> - Deletion: if < 5% and random
> - Imputation: mean/median/mode for numerical/categorical
> - Advanced: KNN imputation, iterative imputation
> 4. Validate impact on model performance
> 5. Document assumptions made

Formula Sheet Creation (30 min):

Create a quick reference with key formulas:

Statistics:
- Standard Error: $\sigma/\sqrt{n}$
- Confidence Interval: $\bar{x} \pm z^*(\sigma/\sqrt{n})$
- t-statistic: $(\bar{x} - \mu)/(s/\sqrt{n})$

ML Metrics:
- Precision: $TP/(TP+FP)$
- Recall: $TP/(TP+FN)$
- F1 Score: $2*(Precision*Recall)/(Precision+Recall)$
- ROC-AUC: Area under ROC curve

Regularization:
- Ridge (L2): $\alpha\sum\beta i^2$
- Lasso (L1): $\alpha\sum|\beta i|$

Final Preparation Tips:

1. **Practice explaining concepts to a rubber duck or mirror**
2. **Time yourself - aim for 2-3 minute explanations**
3. **Prepare 2-3 projects to discuss in detail**
4. **Have questions ready to ask the interviewer**
5. **Practice coding on a whiteboard or paper**

Additional Resources & Tools

Books (Free PDFs Available)

1. **"An Introduction to Statistical Learning"** - Hastie, Tibshirani

2. **"The Elements of Statistical Learning"** - Hastie, Tibshirani, Friedman
3. **"Python Data Science Handbook"** - Jake VanderPlas
4. **"Hands-On Machine Learning"** - Aurélien Géron

Online Platforms

1. **Kaggle Learn**: Free micro-courses with certificates
2. **Coursera**: Audit courses for free (no certificate)
3. **edX**: MIT and Harvard courses, audit option available
4. **Fast.ai**: Practical deep learning courses

Practice Platforms

1. **HackerRank**: Data science challenges
2. **LeetCode**: SQL and algorithm problems
3. **Kaggle Competitions**: Real-world datasets
4. **Google Colab**: Free GPU/TPU access

YouTube Channels

1. **StatQuest with Josh Starmer**: Statistical concepts
2. **3Blue1Brown**: Mathematical intuition
3. **Krish Naik**: End-to-end ML projects
4. **Data School**: Pandas and scikit-learn tutorials

Day-by-Day Schedule Summary

Day 1 Timeline:

- **9:00-12:00**: Stats refresher + practice problems
- **13:00-16:00**: EDA with Pandas (Titanic + Boston Housing)
- **19:00-21:00**: ML theory (algorithms, concepts)

Day 2 Timeline:

- **9:00-12:00**: ML evaluation metrics + end-to-end project
- **13:00-16:00**: Big data concepts + cloud platform overview
- **19:00-21:00**: Mock interview questions + formula review

Success Metrics:

- [ ] Complete 15 statistics problems correctly
- [ ] Finish EDA on 2 datasets with insights
- [ ] Explain 5 ML algorithms confidently
- [ ] Build 1 complete ML pipeline
- [ ] Answer 20 interview questions fluently
- [ ] Create personal formula sheet

Good luck with your interview preparation! Focus on understanding concepts rather than memorizing, and practice explaining your thought process clearly.