Data Science Notes

Essential concepts, code snippets, and best practices — from data cleaning to deployment. A living reference for aspiring and practicing data scientists.

Updated: April 2026 | v2.0

What is Data Science?

Data Science combines statistics, programming, and domain expertise to extract actionable insights from structured and unstructured data.

Statistics & Probability
Foundation of inference & distributions

Programming
Python / R, SQL, Bash

Domain Expertise
Business, healthcare, finance, etc.

Machine Learning
Predictive modeling & AI

Data Science Workflow

Step	Name	Description

Step	Name	Description
1	Problem Definition	Align with business goals, define KPIs
2	Data Collection	CSV, SQL, APIs, web scraping
3	Data Cleaning	Missing values, outliers, duplicates
4	EDA	Exploratory analysis & visual summaries
5	Feature Engineering	Create / transform variables
6	Modeling	Train ML algorithms
7	Evaluation	Test performance & iteration
8	Deployment	API, dashboard, batch inference

Essential Python Libraries

```
import numpy as np          # numerical arrays
import pandas as pd         # dataframes & analysis
import matplotlib.pyplot as plt
import seaborn as sns       # statistical visuals
from sklearn import ...     # preprocessing, modeling
import plotly.express as px # interactive plots
```

Data Cleaning & Preparation

```
# Drop missing values
df.dropna()
```

```
# Fill missing with mean
df.fillna(df.mean())

# Remove duplicates
df.drop_duplicates()

# Change dtype
df['col'] = df['col'].astype('int')

# IQR outlier removal
Q1 = df['col'].quantile(0.25)
Q3 = df['col'].quantile(0.75)
IQR = Q3 - Q1
df = df[(df['col'] > Q1 - 1.5*IQR) & (df['col'] < Q3 + 1.5*IQR)]
```

Exploratory Data Analysis (EDA)

Key questions: distribution, missing patterns, correlations, outliers.

```
# Histogram
df['age'].hist()

# Boxplot
sns.boxplot(x=df['salary'])

# Correlation heatmap
sns.heatmap(df.corr(), annot=True)

# Scatter plot
plt.scatter(df['height'], df['weight'])
```

> **Pro tip:** Always visualize before modeling — a picture tells a thousand p-values.

Statistics Fundamentals

Concept	Meaning
Mean	Average value
Median	Middle value (robust to outliers)
Mode	Most frequent value
Variance / Std	Spread of data around mean
Correlation (r)	Strength of linear relationship (-1 to +1)
P-value	Probability result occurred by chance; $p < 0.05 \rightarrow$ statistically significant

Machine Learning Overview

Supervised Learning

Regression (continuous): Linear Regression, Random Forest, XGBoost.
Classification (categories): Logistic Regression, SVM, Decision Trees.

Unsupervised Learning

Clustering: K-Means, DBSCAN.
Dimensionality reduction: PCA, t-SNE.

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
```

Common Evaluation Metrics

Problem	Metric	Description
Regression	MAE / RMSE	Mean Absolute Error / Root Mean Square Error
Regression	R^2	Variance explained by model (0 to 1)
Classification	Accuracy	% of correct predictions
Classification	Precision	TP / (TP + FP)
Classification	Recall	TP / (TP + FN)
Classification	F1-Score	Harmonic mean of precision & recall
Classification	ROC-AUC	Ability to separate classes (higher = better)

Model Improvement Techniques

- **Cross-validation** (k-fold, Stratified)
- **Hyperparameter tuning** – GridSearchCV / RandomizedSearchCV
- **Feature selection** (mutual info, RFE, regularization)
- **Regularization** – Lasso (L1), Ridge (L2) to reduce overfitting
- **Ensemble methods** – Bagging (Random Forest), Boosting (XGBoost, LightGBM)

Data Visualization Principles

✔ Simplify – remove chartjunk
✔ Label axes clearly
✔ Use accessible color palettes (avoid red-green)
✔ Choose chart type wisely:

- Comparison → Bar chart
- Distribution → Histogram / Boxplot
- Relationship → Scatter plot
- Trend over time → Line chart
- Composition → Stacked bar (avoid 3D pie)

Edward Tufte's rule:
Maximize data-ink ratio. Every non-data pixel is noise.

Pandas Quick Reference

Task	Code
Load CSV	`pd.read_csv('file.csv')`
First 5 rows	`df.head()`
Summary stats	`df.describe()`
DataFrame info	`df.info()`
Select column	`df['col']`
Filter rows	`df[df['col'] > 10]`
Group by & aggregate	`df.groupby('category')['value'].mean()`
Merge two DFs	`pd.merge(df1, df2, on='key')`

Real-World Data Science Tips

80% of time = data cleaning
Never skip exploratory checks.

Visualize before modeling
Reveals outliers, patterns, assumptions.

Start simple
Baseline model (linear/logistic) then iterate.

Set random seeds
Reproducibility is professional.

Document everything
Jupyter notebooks + markdown.

Learn SQL
Most real data lives in warehouses.

Free Learning Resources

Kaggle Learn Google Data Analytics StatQuest

Python for Data Analysis Scikit-learn docs

Stay curious, always question your data, and share insights.