

خطة دراسة الإحصاء الوصفي - الفيديو التأسيسي

معلومات الفيديو

Descriptive Statistics Basics || العنوان: الاحصاء الوصفية كاملة في فيديو واحد

الرابط: https://www.youtube.com/watch?v=8wwPwlueoDs&list=PLtsZ69x5q-X_MJj_iwBwpJaLg_C6JGiWW

المدة: 5 ساعات و 30 دقيقة

القناة: Elgohary AI

المشاهدات: 58,858 مشاهدة

تاريخ النشر: 4 يونيو 2022

فهرس المحتويات التفصيلي

الوقت	الموضوع	الأهمية للمشروع
00:00	مقدمة عن مبادئ علم الاحصاء	☆☆☆☆☆
03:27	Cases, Variables, and levels of measurements	☆☆☆☆☆
14:07	Data Matrix and frequency table	☆☆☆☆☆
21:15	Graphs and Shapes of distributions	☆☆☆☆☆
29:11	Measures of Central Tendency	☆☆☆☆☆
38:36	Interquartile Range	☆☆☆☆
49:01	Variance and Standard deviation	☆☆☆☆☆
56:50	Z-Score	☆☆☆☆☆
01:06:14	مثال على ما سبق	☆☆☆☆
01:17:12	Correlation between two variables	☆☆☆☆☆
01:27:50	Pearson's R	☆☆☆☆☆
01:38:02	Regression	☆☆☆☆
02:03:27	Correlation vs. Causation	☆☆☆☆
02:11:56	Example on Contingency table	☆☆☆☆
02:16:32	Example of Regression Analysis	☆☆☆
02:26:58	Randomness and Probability	☆☆☆
02:34:39	Probabilities and Tree Diagram	☆☆☆
02:45:04	Probabilities and Sets	☆☆
03:06:41	Joint, marginal, and conditional probability	☆☆
03:16:44	Independence between Random events	☆☆

الوقت	الموضوع	الأهمية للمشروع
03:22:27	Decision Tree and Bayes' law	☆☆☆
03:34:41	Probability and cumulative distribution	☆☆☆
03:47:58	The mean and the variance of random variable	☆☆☆
04:06:30	The Normal Distribution	☆☆☆☆
04:17:09	The standard normal distribution	☆☆☆☆
04:29:07	The binomial distribution	☆☆
04:38:45	Population vs. sample	☆☆☆☆☆
04:52:59	The sampling distribution of the sample mean	☆☆☆
05:00:04	The central limit theorem	☆☆☆
05:08:51	Population vs. sample vs. sampling distribution	☆☆☆
05:19:05	The sampling distribution of the sample proportion	☆☆
05:24:56	Final example	☆☆☆

✓ لماذا هذا الفيديو مناسب تماماً؟

🎯 المطابقة مع مشروع السكري:

- 1. شامل ومتكامل: يغطي كل الإحصاء الوصفي المطلوب
- 2. باللغة العربية: يسهل الفهم والاستيعاب
- 3. تطبيقي: يحتوي على أمثلة عملية
- 4. مناسب للمبتدئين: يبدأ من الأساسيات
- 5. يغطي أول 4 مواضيع من دليل المفاهيم الرياضية

📊 المواضيع المطابقة مع المشروع:

(الإحصاء الوصفي (01:06:14-00:00):

- ✓ تحليل خصائص المرضى (العمر، الجنس، مستوى السكر
- ✓ حساب المتوسطات والانحرافات المعيارية
- ✓ تحليل التوزيعات
- ✓ Z-Score اكتشاف القيم الشاذة باستخدام

(التصور والرسوم البيانية (29:11-21:15):

- ✓ Histograms لتوزيع الأعمار
- ✓ Box plots لمقارنة مستويات HbA1c
- ✓ Bar charts للمتغيرات الفئوية

(الارتباط والانحدار (02:16:32-01:17:12):

- ☒ تحليل العلاقة بين العمر والمضاعفات
- ☒ ارتباط مستوى السكر بالمضاعفات
- ☒ Pearson correlation coefficient

📊 خطة الدراسة التفصيلية

(الأسبوع الأول: الأساسيات (7 أيام

(اليوم الأول (1.5 ساعة

مقدمة: Variables + Data Matrix + الموضوع:

الوقت: 00:00 - 21:15

الهدف: فهم أنواع البيانات والمتغيرات

المهام:

- ☐ مشاهدة الجزء الأول (21 دقيقة)
- ☐ تطبيق على بيانات المشروع
- ☐ السكري dataset تحديد نوع كل متغير في

كود التطبيق:

```
import pandas as pd
import numpy as np

# قراءة البيانات
df = pd.read_excel('data/Data_DM.xlsx', sheet_name='Data_Set')

# تحديد أنواع المتغيرات
print("أنواع المتغيرات:")
print(df.dtypes)

# المتغيرات النوعية (Categorical)
categorical_vars = ['Gender', 'Hypertension', 'Heart_Disease', 'Smoking_History']

# المتغيرات الكمية (Numerical)
numerical_vars = ['Age', 'BMI', 'HbA1c_level', 'Blood_glucose_level']

print(f"المتغيرات النوعية: {categorical_vars}")
print(f"المتغيرات الكمية: {numerical_vars}")
```

(اليوم الثاني (1.5 ساعة

الموضوع: Graphs and Shapes of distributions

الوقت: 21:15 - 29:11

الهدف: فهم أشكال التوزيعات والرسوم البيانية

المهام:

- ☐ مشاهدة الجزء (8 دقائق)
- ☐ رسم توزيعات المتغيرات الرئيسية
- ☐ تحليل شكل كل توزيع

كود التطبيق:

```
import matplotlib.pyplot as plt
import seaborn as sns

# إعداد الرسوم البيانية
plt.figure(figsize=(15, 10))

# توزيع الأعمار
plt.subplot(2, 3, 1)
df['Age'].hist(bins=20, alpha=0.7)
plt.title('Age Distribution')
plt.xlabel('Age')
plt.ylabel('Frequency')

# توزيع مستوى السكر
plt.subplot(2, 3, 2)
df['HbA1c_level'].hist(bins=20, alpha=0.7, color='green')
plt.title('HbA1c Level Distribution')

# BMI توزيع
plt.subplot(2, 3, 3)
df['BMI'].hist(bins=20, alpha=0.7, color='red')
plt.title('BMI Distribution')

# الجنس
plt.subplot(2, 3, 4)
df['Gender'].value_counts().plot(kind='bar')
plt.title('Gender Distribution')

# ضغط الدم
plt.subplot(2, 3, 5)
df['Hypertension'].value_counts().plot(kind='bar')
plt.title('Hypertension Distribution')

# المضاعفات
plt.subplot(2, 3, 6)
df['Complications'].value_counts().plot(kind='bar')
plt.title('Complications Distribution')

plt.tight_layout()
plt.show()
```

(اليوم الثالث (1.5 ساعة

الموضوع: Measures of Central Tendency

الوقت: 29:11 - 38:36

الهدف: فهم المتوسط والوسيط والمنوال

المهام:

- ☐ مشاهدة الجزء (9 دقائق)
- ☐ حساب مقاييس النزعة المركزية لكل متغير
- ☐ مقارنة النتائج وتفسيرها

كود التطبيق:

```
# مقاييس النزعة المركزية
def central_tendency_analysis(column_name):
    data = df[column_name].dropna()

    mean_val = data.mean()
    median_val = data.median()
    mode_val = data.mode().iloc[0] if not data.mode().empty else "No mode"

    print(f"\n=== {column_name} ===")
    print(f"المتوسط (Mean): {mean_val:.2f}")
    print(f"الوسيط (Median): {median_val:.2f}")
    print(f"المنوال (Mode): {mode_val}")

# تفسير النتائج
if mean_val > median_val:
    print("التوزيع: منحرف إيجابياً (Right Skewed)")
elif mean_val < median_val:
    print("التوزيع: منحرف سلبياً (Left Skewed)")
else:
    print("التوزيع: متماثل (Symmetric)")

# تطبيق على المتغيرات الرئيسية
for var in ['Age', 'BMI', 'HbA1c_level', 'Blood_glucose_level']:
    central_tendency_analysis(var)
```

(اليوم الرابع (1.5 ساعة

الموضوع: Interquartile Range

الوقت: 38:36 - 49:01

الهدف: فهم الأرباع والمدى الرباعي

المهام:

- ☐ مشاهدة الجزء (10 دقائق)
- ☐ حساب Q1, Q2, Q3, IQR
- ☐ IQR اكتشاف القيم الشاذة باستخدام

كود التطبيق:

```
def quartile_analysis(column_name):
    data = df[column_name].dropna()

    Q1 = data.quantile(0.25)
    Q2 = data.quantile(0.50) # الوسيط
    Q3 = data.quantile(0.75)
    IQR = Q3 - Q1

    # حدود القيم الشاذة
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR

    # القيم الشاذة
    outliers = data[(data < lower_bound) | (data > upper_bound)]

    print(f"\n=== {column_name} - Quartile Analysis ===")
    print(f"Q1 (الربع الأول): {Q1:.2f}")
    print(f"Q2 (الوسيط): {Q2:.2f}")
    print(f"Q3 (الربع الثالث): {Q3:.2f}")
    print(f"IQR (المدى الربعي): {IQR:.2f}")
    print(f"الحد الأدنى للقيم العادية: {lower_bound:.2f}")
    print(f"الحد الأعلى للقيم العادية: {upper_bound:.2f}")
    print(f"عدد القيم الشاذة: {len(outliers)}")

    # رسم Box plot
    plt.figure(figsize=(8, 6))
    df.boxplot(column=column_name)
    plt.title(f'Box Plot for {column_name}')
    plt.ylabel(column_name)
    plt.show()

    # تطبيق على المتغيرات
    for var in ['Age', 'BMI', 'HbA1c_level']:
        quartile_analysis(var)
```

(اليوم الخامس (1.5 ساعة):

الموضوع: Variance and Standard deviation

الوقت: 49:01 - 56:50

الهدف: فهم التباين والانحراف المعياري

المهام:

- ☐ مشاهدة الجزء (8 دقائق)
- ☐ حساب التباين والانحراف المعياري
- ☐ تفسير النتائج ومقارنة المتغيرات

كود التطبيق:

```
def variance_analysis(column_name):
    data = df[column_name].dropna()

    mean_val = data.mean()
    variance = data.var()
    std_dev = data.std()

    # معامل الاختلاف (Coefficient of Variation)
    cv = (std_dev / mean_val) * 100

    print(f"\n=== {column_name} - Measures of Dispersion ===")
    print(f"المتوسط: {mean_val:.2f}")
    print(f"التباين (Variance): {variance:.2f}")
    print(f"الانحراف المعياري (Std Dev): {std_dev:.2f}")
    print(f"معامل الاختلاف: {cv:.2f}%")

    # تفسير معامل الاختلاف
    if cv < 15:
        print("التشتت: منخفض")
    elif cv < 35:
        print("التشتت: متوسط")
    else:
        print("التشتت: عالي")

    # تطبيق على جميع المتغيرات الرقمية
    numerical_vars = ['Age', 'BMI', 'HbA1c_level', 'Blood_glucose_level']
    for var in numerical_vars:
        variance_analysis(var)

    # مقارنة التشتت بين المتغيرات
    plt.figure(figsize=(12, 8))
    df[numerical_vars].boxplot()
    plt.title('مقارنة التشتت بين المتغيرات')
    plt.xticks(rotation=45)
    plt.show()
```

(اليوم السادس (1.5 ساعة

الموضوع: Z-Score

الوقت: 01:06:14 - 56:50

الهدف: فهم النتيجة المعيارية وتطبيقها

المهام:

- ☐ مشاهدة الجزء (9 دقائق)
- ☐ لكل متغير Z-Score حساب
- ☐ Z-Score اكتشاف القيم الشاذة باستخدام

كود التطبيق:

```

from scipy import stats

def zscore_analysis(column_name):
    data = df[column_name].dropna()

    # حساب Z-Score
    z_scores = stats.zscore(data)

    # إضافة Z-Score إلى DataFrame
    df[f'{column_name}_zscore'] = np.nan
    df.loc[data.index, f'{column_name}_zscore'] = z_scores

    # القيم الشاذة (|Z-Score| > 3)
    extreme_outliers = abs(z_scores) > 3
    moderate_outliers = (abs(z_scores) > 2) & (abs(z_scores) <= 3)

    print(f"\n=== {column_name} - Z-Score Analysis ===")
    print(f"عدد القيم الشاذة الشديدة (|Z| > 3): {sum(extreme_outliers)}")
    print(f"عدد القيم الشاذة المتوسطة (|Z| <= 3): {sum(moderate_outliers)}")
    print(f"النسبة المئوية للقيم العادية: {(len(z_scores) - sum(abs(z_scores) > 2)) / len(z_scores) * 100):.1f}%")

    # رسم توزيع Z-Scores
    plt.figure(figsize=(10, 6))
    plt.subplot(1, 2, 1)
    plt.hist(data, bins=20, alpha=0.7, color='blue')
    plt.title(f'Original {column_name} Distribution')
    plt.xlabel(column_name)

    plt.subplot(1, 2, 2)
    plt.hist(z_scores, bins=20, alpha=0.7, color='red')
    plt.title(f'{column_name} Z-Scores Distribution')
    plt.xlabel('Z-Score')
    plt.axvline(x=-2, color='orange', linestyle='--', label='Z = -2')
    plt.axvline(x=2, color='orange', linestyle='--', label='Z = 2')
    plt.axvline(x=-3, color='red', linestyle='--', label='Z = -3')
    plt.axvline(x=3, color='red', linestyle='--', label='Z = 3')
    plt.legend()

    plt.tight_layout()
    plt.show()

# تطبيق على المتغيرات الرئيسية
for var in ['Age', 'BMI', 'HbA1c_level']:
    zscore_analysis(var)

```

(اليوم السابع: مراجعة وتطبيق شامل (2 ساعة)

الهدف: مراجعة كل ما تم تعلمه وإنشاء تقرير شامل

المهام:

- ☐ مراجعة جميع المفاهيم
- ☐ إنشاء تقرير إحصائي شامل للبيانات
- ☐ حل الأسئلة والتمارين

كود التطبيق الشامل:

```
def comprehensive_descriptive_analysis():
    """تحليل إحصائي وصفي شامل لبيانات السكري"""

    print("=" * 50)
    print("تقرير الإحصاء الوصفي الشامل - بيانات مرضى السكري")
    print("=" * 50)

    # 1. معلومات عامة عن البيانات
    print(f"\n1. معلومات عامة:")
    print(f"عدد الحالات: {len(df)}")
    print(f"عدد المتغيرات: {len(df.columns)}")
    print(f"البيانات المفقودة: {df.isnull().sum().sum()}")

    # 2. الإحصائيات الوصفية للمتغيرات الرقمية
    print(f"\n2. الإحصائيات الوصفية:")
    numerical_summary = df.select_dtypes(include=[np.number]).describe()
    print(numerical_summary.round(2))

    # 3. توزيع المتغيرات الفئوية
    print(f"\n3. توزيع المتغيرات الفئوية:")
    categorical_vars = df.select_dtypes(include=['object']).columns
    for var in categorical_vars:
        print(f"\n{var}:")
        print(df[var].value_counts())
        print(f"النسب المئوية:")
        print((df[var].value_counts(normalize=True) * 100).round(1))

    # 4. تحليل المضاعفات
    print(f"\n4. تحليل المضاعفات:")
    complications_rate = df['Complications'].mean() * 100
    print(f"معدل حدوث المضاعفات: {complications_rate:.1f}%")

    # 5. المقارنات حسب المضاعفات
    print(f"\n5. مقارنة المتغيرات حسب وجود المضاعفات:")
    comparison = df.groupby('Complications')[numerical_vars].agg(['mean', 'std'])
    print(comparison.round(2))

    # 6. مصفوفة الارتباط
    print(f"\n6. مصفوفة الارتباط:")
    correlation_matrix = df[numerical_vars + ['Complications']].corr()
    print(correlation_matrix.round(3))

    # رسم مصفوفة الارتباط
    plt.figure(figsize=(10, 8))
    sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', center=0)
    plt.title('مصفوفة الارتباط - متغيرات مرضى السكري')
```

```
plt.show()
```

```
# تشغيل التحليل الشامل  
comprehensive_descriptive_analysis()
```

(الأسبوع الثاني: الارتباط والانحدار (7 أيام

(اليوم الثامن (1.5 ساعة

الموضوع: Correlation between two variables

الوقت: 01:17:12 - 01:27:50

الهدف: فهم مفهوم الارتباط وقياسه

المهام:

- ☐ مشاهدة الجزء (10 دقائق)
- ☐ حساب معاملات الارتباط
- ☐ تفسير قوة واتجاه العلاقات

كود التطبيق:

```
def correlation_analysis():  
    """تحليل الارتباط بين المتغيرات"""  
  
    # حساب معاملات الارتباط  
    numerical_vars = ['Age', 'BMI', 'HbA1c_level', 'Blood_glucose_level']  
    correlation_matrix = df[numerical_vars].corr()  
  
    print("مصفوفة الارتباط:")  
    print(correlation_matrix.round(3))  
  
    # تحليل الارتباط مع المضاعفات  
    print(f"\nارتباط المتغيرات مع المضاعفات:")  
    for var in numerical_vars:  
        corr = df[var].corr(df['Complications'])  
        print(f"{var}: {corr:.3f}")  
  
    # تفسير قوة الارتباط  
    if abs(corr) < 0.1:  
        strength = "ضعيف جداً"  
    elif abs(corr) < 0.3:  
        strength = "ضعيف"  
    elif abs(corr) < 0.5:  
        strength = "متوسط"  
    elif abs(corr) < 0.7:  
        strength = "قوي"  
    else:  
        strength = "قوي جداً"  
  
    direction = "إيجابي" if corr > 0 else "سلبي"
```

```

print(f" - القوة : {strength}, الاتجاه : {direction}")

# رسم المخططات المبعثرة
fig, axes = plt.subplots(2, 2, figsize=(15, 12))
axes = axes.ravel()

for i, var in enumerate(numerical_vars):
    axes[i].scatter(df[var], df['Complications'], alpha=0.6)
    axes[i].set_xlabel(var)
    axes[i].set_ylabel('Complications')
    axes[i].set_title(f'والمضاعفات {var} العلاقة بين')

    # إضافة خط الاتجاه
    z = np.polyfit(df[var].dropna(), df['Complications'][df[var].notna()], 1)
    p = np.poly1d(z)
    axes[i].plot(df[var], p(df[var]), "r--", alpha=0.8)

plt.tight_layout()
plt.show()

correlation_analysis()

```

(اليوم التاسع (1.5 ساعة

الموضوع: Pearson's R

الوقت: 01:38:02 - 01:27:50

الهدف: فهم معامل ارتباط بيرسون بالتفصيل

المهام:

- ☐ مشاهدة الجزء (10 دقائق)
- ☐ Python يدوياً ومقارنته مع Pearson's R حساب
- ☐ اختبار معنوية الارتباط

كود التطبيق:

```

from scipy.stats import pearsonr

def pearson_detailed_analysis(var1, var2):
    """تحليل مفصل لمعامل ارتباط بيرسون"""

    # تنظيف البيانات
    data = df[[var1, var2]].dropna()
    x = data[var1]
    y = data[var2]

    # حساب p-value لمعامل الارتباط والـ
    r, p_value = pearsonr(x, y)

    # حساب معامل الثقة
    n = len(data)

```

```

r_squared = r ** 2

print(f"\n=== تحليل الارتباط بين {var1} و {var2} ===")
print(f"معامل الارتباط (r): {r:.4f}")
print(f"معامل التحديد (r²): {r_squared:.4f}")
print(f"P-value: {p_value:.6f}")
print(f"حجم العينة: {n}")

# تفسير النتائج
if p_value < 0.001:
    significance = "عالية جداً (p < 0.001)"
elif p_value < 0.01:
    significance = "عالية (p < 0.01)"
elif p_value < 0.05:
    significance = "معنوية (p < 0.05)"
else:
    significance = "غير معنوية (p >= 0.05)"

print(f"المعنوية الإحصائية: {significance}")
print(f"النسبة المئوية للتباين المفسر: {r_squared * 100:.1f}%")

# رسم العلاقة
plt.figure(figsize=(10, 6))
plt.scatter(x, y, alpha=0.6, color='blue')

# خط الانحدار
z = np.polyfit(x, y, 1)
p = np.poly1d(z)
plt.plot(x, p(x), "r-", linewidth=2, label=f'r = {r:.3f}')

plt.xlabel(var1)
plt.ylabel(var2)
plt.title(f'الارتباط بين {var1} و {var2}')
plt.legend()
plt.grid(True, alpha=0.3)
plt.show()

# تحليل العلاقات المهمة
important_pairs = [
    ('Age', 'HbA1c_level'),
    ('BMI', 'Blood_glucose_level'),
    ('HbA1c_level', 'Blood_glucose_level'),
    ('Age', 'Complications')
]

for var1, var2 in important_pairs:
    pearson_detailed_analysis(var1, var2)

```

الأيام 10-14: الانحدار والمفاهيم المتقدمة

[□□□□□□□□ □□□□□□ □□□□□□...]

أدوات التطبيق العملي

البرامج المطلوبة:

- ☐ Python 3.8+
- ☐ Jupyter Notebook أو VS Code
- ☐ مكتبات Python:

```
pip install pandas numpy matplotlib seaborn scipy
```

الملفات المطلوبة:

- ☐ بيانات مشروع السكري: [data/Data_DM.xlsx](#)
- ☐ دفتر الملاحظات: [notebooks/descriptive_stats_practice.ipynb](#)

نصائح للاستفادة القصوى

أثناء المشاهدة:

1. **توقف وطبق:** عند كل مفهوم جديد، توقف وطبقه على بياناتك.
2. **اكتب ملاحظات:** سجل النقاط المهمة والصيغ الرياضية.
3. **كرر الأجزاء الصعبة:** لا تتردد في إعادة المشاهدة.
4. **استخدم السرعة المناسبة:** ابطئ في الأجزاء المعقدة.

بعد كل جلسة:

1. **راجع الكود:** تأكد من فهم كل سطر كتيته.
2. **اختبر فهمك:** اطرح أسئلة على نفسك.
3. **اربط بالمشروع:** كيف يفيد هذا المفهوم في تحليل بيانات السكري؟
4. **سجل الأسئلة:** اكتب أي استفسارات للبحث عنها لاحقاً.

نصائح تقنية:

- للتطبيق التفاعلي **Jupyter Notebook** استخدم
- **احفظ الكود** في ملفات منفصلة لكل موضوع
- **لعملك بانتظام backup** اعمل
- **شارك كودك** مع زملائك للمراجعة

أهداف التعلم لكل أسبوع

نهاية الأسبوع الأول:

يجب أن تكون قادراً على:

- ☐ dataset تحديد نوع أي متغير في أي
- ☐ حساب جميع مقاييس النزعة المركزية والتشتت
- ☐ أساسية للبيانات visualizations إنشاء

- ☐ اكتشاف القيم الشاذة بطرق مختلفة
- ☐ تفسير النتائج الإحصائية بشكل صحيح

نهاية الأسبوع الثاني:

يجب أن تكون قادراً على:

- ☐ حساب وتفسير معاملات الارتباط
- ☐ بناء نماذج انحدار بسيطة
- ☐ التمييز بين الارتباط والسببية
- ☐ تحليل العلاقات بين المتغيرات
- ☐ إنشاء تقارير إحصائية شاملة

☒ قياس التقدم

اختبار نهاية الأسبوع الأول:

1. احسب المتوسط والانحراف المعياري لعمر المرضى
2. HbA1c لتوزيع مستوى histogram ارسم
3. BMI باستخدام IQR و Z-Score اكتشاف القيم الشاذة في
4. فسر النتائج بجملة مفيدة

اختبار نهاية الأسبوع الثاني:

1. احسب معامل الارتباط بين العمر ومستوى السكر
2. اختبر معنوية هذا الارتباط
3. بناء نموذج انحدار للتنبؤ بمستوى السكر من العمر
4. اكتب تقريراً يلخص جميع النتائج

روابط مفيدة إضافية

مراجع سريعة:

- [Pandas Cheat Sheet](#)
- [Matplotlib Gallery](#)
- [Seaborn Examples](#)

قنوات تعليمية مفيدة:

- **StatQuest:** شرح مبسط للمفاهيم الإحصائية
- **3Blue1Brown:** الرياضيات بصرياً
- **Khan Academy:** دروس تفاعلية

مجتمعات للمساعدة:

- **Stack Overflow:** للأسئلة التقنية
- **Reddit r/statistics:** مناقشات إحصائية
- **Kaggle Learn:** دورات مجانية تطبيقية

✓ قائمة المراجعة النهائية

قبل البدء:

- ☐ تحميل البيانات وتجهيز البيئة البرمجية
- ☐ pandas و Python مراجعة أساسيات
- ☐ تحديد الأهداف اليومية والأسبوعية

أثناء الدراسة:

- ☐ متابعة الجدول الزمني المحدد
- ☐ تطبيق كل مفهوم عملياً
- ☐ كتابة الملاحظات والملاحظات

بعد الانتهاء:

- ☐ مراجعة شاملة لجميع المفاهيم
- ☐ حل تمارين إضافية
- ☐ الاستعداد للمرحلة التالية (Machine Learning)

تاريخ الإنشاء: أغسطس 2025

آخر تحديث: أغسطس 2025

المؤلف: مساعد التعلم الذكي

هذا المستند هو ملخص لبرنامج دراسي يهدف إلى تعزيز فهمك لمفاهيم التعلم الآلي وتحليل البيانات باستخدام Python و pandas. تم تصميمه ليكون مرجعاً سريعاً يمكنك الرجوع إليه أثناء دراستك. نأمل أن يساعدك هذا البرنامج على تحقيق أهدافك التعليمية.