

جمهورية العراق — وزارة الصحة

مديرية التشخيص • برنامج التدريب المهني

بايثون والذكاء الاصطناعي للتشخيص الطبي

من البيانات المخبرية إلى نماذج التنبؤ التشخيصي —
من دون برمجة تقليدية



د. حسام صلاح مهدي

مدير الدورة • 10 أيام • 90 دقيقة يومياً

المرحلة الأولى — إتقان البيانات

- 1 الأسس وهياكل البيانات الصحية
- 2 استيراد السجلات وأول تحليل استكشافي
- 3 تنظيف البيانات الطبية

المرحلة الثانية — الرؤية والبصيرة

- 4 الفلتر المنطقية وتحليلات العمليات السريرية
- 5 التصوير البصري للبيانات

المرحلة الثالثة — تعلّم الآلة

- 6 هندسة الميزات
- 7 أسس تعلّم الآلة (التقسيم والمعايرة)
- 8 نماذج التصنيف التشخيصية
- 9 الحساسية والنوعية وROC وAUC

المرحلة الرابعة — التشغيل

- 10 مسارات العمل والأخلاقيات ومشروع التخرج

اليوم 1

الأسس التقنية وهياكل البيانات الصحية

من الحدس السريري إلى التشخيص المستند إلى البيانات — جلستك الأولى في بايثون

90 دقيقة • 20 نظري — 40 تطبيق موجّه — 30 تحدّ سريري

خارطة طريق اليوم وأهدافه

90 دقيقة

- 20 دقيقة — نظري: التحول المستند إلى البيانات، بايثون، Colab، المكتبات
- 40 دقيقة — تطبيق موجّه: المتغيرات، سلاسل f، القوائم، القواميس
- 30 دقيقة — تحدّي: ملف مريض رقمي كامل

ستكون قادراً على

- تشغيل دفاتر Colab ومشاركتها
- مطابقة bool و str و float و int مع البيانات السريّة
- تلخيص سلاسل مختبرية بدوال القوائم الجاهزة — بلا حلقات
- نمذجة مريض واحد بقاموس (متداخل)، وتجميع ردهة كاملة كقائمة قواميس

التحول المستند إلى البيانات في التشخيص

حلقة التغذية الراجعة التشخيصية

نقيس ← نسجّل ← نحلّل ← نقرّر ← نقيس من جديد

- قيمة الهيموغلوبين على الورق تجيب عن سؤال واحد لطبيب واحد
- القيمة نفسها في سجل مهيكّل تجيب عن آلاف الأسئلة، وإلى الأبد
- الورق يكسر الحلقة عند التسجيل: لا استعلامات، ولا ضبط جودة، ولا اتجاهات
- هذه الدورة تقوّي التسجيل والتحليل حتى تستند القرارات إلى الدليل

انضباط الدورة

كل تحليل يعمل على مجموعة كاملة دفعة واحدة — القوائم اليوم، وأعمدة DataFrame غداً. لا حلقات تكرارية ولا if/else في الشيفرة المُدرّسة.

لماذا بايثون — ولماذا Google Colab

لماذا بايثون

- مفتوحة المصدر ومقروءة: شيفرة مجانية يستطيع الفريق السريري تدقيقها
- منظومة علمية: NumPy pandas و scikit-learn — لغة الذكاء الاصطناعي الطبي المنشور

لماذا Colab

- يعمل في المتصفح — صفر تثبيت؛ وبيئة التشغيل السحابية هي التي تحسب
- الخلايا: شيفرة + مخرجات + شرح معاً؛ ويُشارك كمستند (عرض/تحرير)
- المنظومة العلمية مثبتة مسبقاً؛ وفي غيره: requirements.txt + pip

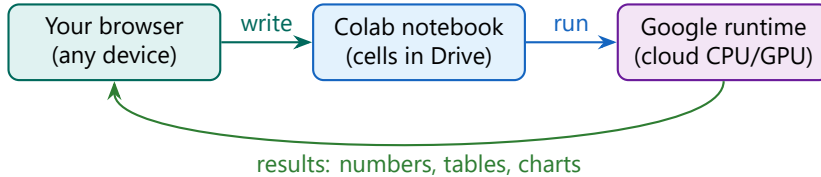
أنواع بايثون هي أنواع البيانات السريية

النوع	المعنى	مثال سريي	الصيغة
int	عدد صحيح	العمر، مدة الرقود	58
float	عدد عشري	الكرياتينين، Ct HbA1c	1.4
str	نص	معرّف المريض، التشخيص	"MOH-00042"
bool	راية نعم/لا	مدخّن، PCR إيجابي	True

أكلف أخطاء الجداول الإلكترونية

القيمة "58" نص لا عدد — ولا متوسط لها. علامات الاقتباس للمعرّفات والنصوص، وأبداً ليست للقياسات.

سير العمل في Colab



- بيئة التشغيل مؤقتة — نص الدفتر وشيفرته محفوظان بأمان في Drive
- حتى الحاسوب المتواضع يحلل مجموعات بيانات كبيرة: السحابة هي التي تحسب

المتغيرات والأنواع وسطر التقرير بسلسلة f

```

patient_id = "MOH-00042"    # str
age = 58                    # int
creatinine_mgdl = 1.4        # float
smoker = True                # bool

name = "Ali Hassan"
glucose_mgdl = 182.0
hba1c_pct = 8.14
print(f"{patient_id} | {name} | {age} y | "
      f"glucose {glucose_mgdl:.0f} mg/dL | HbA1c {hba1c_pct:.1f} %")

```

```
MOH-00042 | Ali Hassan | 58 y | glucose 182 mg/dL | HbA1c 8.1 %
```

- 8.1% : يضبط المنازل العشرية — تنسيق بمستوى التقارير
- غلوكوز 182 مع HbA1c 8.1%: السطر يُقرأ فعلياً كورقة إحالة

سلسلة قراءات الغلوكوز ودوالها الجاهزة — بلا حلقة

```
glucose_readings = [148, 162, 155, 171, 149, 180, 158]

print("Number of readings:", len(glucose_readings))
print("Lowest :", min(glucose_readings), "mg/dL")
print("Highest:", max(glucose_readings), "mg/dL")
mean_glucose = sum(glucose_readings) / len(glucose_readings)
print(f"Mean    : {mean_glucose:.1f} mg/dL")
```

```
Number of readings: 7
Lowest : 148 mg/dL
Highest: 180 mg/dL
Mean    : 160.4 mg/dL
```

- كل قراءة تتجاوز عتبة 126 ملغ/دل: فرط سكر دم مستمر، أُجيب عنه في خمسة أسطر

آخر حلقة تكرارية ستراها هنا

```
# ANTI-PATTERN -- shown once, then banned
total = 0
for g in glucose_readings:
    total = total + g
mean_loop = total / len(glucose_readings)
# Course standard -- whole-collection, one line
mean_builtin = sum(glucose_readings) / len(glucose_readings)
import numpy as np
mean_numpy = np.mean(glucose_readings)
print(mean_loop, mean_builtin, mean_numpy)
```

```
160.42857142857142 160.42857142857142 160.42857142857142
```

قاعدة الدورة

إجابات متطابقة، لكن الحلقة تخفي القصد. لا for ولا while ولا if/else في الشيفرة المُدرّسة

مريض واحد في قاموس (متداخل)

```
patient = {"patient_id": "MOH-00042", "name": "Ali Hassan",  
           "age": 58, "sex": "M", "smoker": True}  
  
print(patient["name"], "is", patient["age"], "years old")  
patient["governorate"] = "Baghdad"      # add a field  
patient["cbc"] = {"wbc": 12.7, "hgb": 10.9, "plt": 214}  
print("Hgb:", patient["cbc"]["hgb"], "g/dL")  
anemic = patient["cbc"]["hgb"] < 13      # male cut-off  
print("Anemia flag:", anemic)
```

```
Ali Hassan is 58 years old  
Hgb: 10.9 g/dL  
Anemia flag: True
```

الردّة قائمة قواميس — التشويقة

```
ward = [
    {"patient_id": "MOH-00042", "age": 59, "sex": "M", "glucose": 182.0},
    {"patient_id": "MOH-00107", "age": 34, "sex": "F", "glucose": 94.0},
    {"patient_id": "MOH-00311", "age": 71, "sex": "M", "glucose": 143.0},
]
print("Patients in ward:", len(ward))

import pandas as pd
print(pd.DataFrame(ward))          # explained tomorrow
```

```
Patients in ward: 3
  patient_id  age sex  glucose
0  MOH-00042   59  M    182.0
1  MOH-00107   34  F     94.0
2  MOH-00311   71  M    143.0
```

التحدي السريري — ملف المريض الرقمي في العناية المركزة

السيناريو (تجربة العناية المركزة في البصرة)

زينب كاظم، MOH-00873، 63 عاماً، أنثى. الحرارة 38.4، الضغط الانقباضي 148، النبض 92/101/88/110/97/104، الوزن 82.0 كغم، الطول 1.74 م. صورة الدم: WBC 13.8، Hgb 10.4، Hct 32.1، Plt 512. فحص RT-PCR: إيجابي، Ct 18.4 (ORF1ab).

1 احسب $bmi = weight_kg / height_m^{**2}$

2 ابن قاموساً متداخلاً: demographics / vitals / cbc / rtPCR

3 أدرج مؤشر كتلة الجسم في vitals؛ واقرأ الهيموغلوبين وقيمة Ct بالوصول المتسلسل

4 إحصاءات النبض بالدوال len و min و max و sum؛ والراية $max(pulse) > 100$

5 اطبع ملخصاً سريرياً من خمسة أسطر بسلسلة f، وبلغ عن اكتمال السجل باستخدام len

القواعد

متغيرات وقوائم وقواميس وسلاسل f فقط — لا حلقات ولا جملاً شرطية.

خلاصات اليوم

- السجلات المهيكلة تحوّل النتائج المتفرقة إلى دليل قابل للاستعلام — والدورة تبني هذا المسار
- بايثون: مجانية، قابلة للتدقيق، وبأعمق منظومة علمية؛ Colab: دفاتر سحابية بلا تثبيت
- int للعمر، float للكرياتينين، str لمعرّف المريض، bool لحالة التدخين — والأرقام بين علامتي اقتباس نصوص
- سلاسل f مع .1f : تبني أسطراً بمستوى التقارير
- الدوال الجاهزة len ming maxg sumg sortedg تلخّص السلاسل — بلا حلقات، إلى الأبد
- القاموس سجل مريض (شكل JSON للسجلات الصحية الإلكترونية)؛ وقائمة القواميس ردهة
- pd.DataFrame(ward): سطر واحد، جدول واحد

غداً — اليوم الثاني

pd.read_csv تحمّل 1,200 سجل مريض حقيقي دفعة واحدة: استيراد السجلات الطبية وأول تحليل استكشافي.

اليوم 2

استيراد السجلات الطبية وأول تحليل استكشافي

أطر بيانات Pandas — مجهر البيانات المختبرية

90 دقيقة • 20 نظري — 40 تطبيق موجّه — 30 تحدّ سريري

جدول الأعمال والأهداف

دقائق اليوم التسعون

- تحميل أي تصدير LIS بالأمر `pd.read_csv()`
- قراءة `info()` و `describe()` قراءة سريرية
- قياس معدل الانتشار بالأمر `value_counts()`
- عدّ المرضى فوق عتبة بقناع منطقي
- كتابة مذكرة جودة بيانات من بيانات خام

- 20 دقيقة — نظرية: ما بعد Excel، وتشرح إطار البيانات، وبروتوكول النظرة الأولى
- 40 دقيقة — تطبيق موجّه على `diabetes_dataset.csv` و `cbc_results.csv`
- 30 دقيقة — تحدّ: استخراج ملف `chemistry_panel.csv` للمدير العام

لماذا ينهار Excel

أربعة أنماط فشل صامتة

- **حدود الصفوف:** تتوقف ورقة العمل عند 1,048,576 صفاً (وصيغة x1s . القديمة: 65,536) — الصفوف الزائدة تتلاشى دون إنذار
- **تحويل الأنواع:** المعرّف 00042 يصبح 42؛ وأسماء جينات صارت تواريخ في أوراق منشورة
- **لا قابلية لإعادة الإنتاج:** سلسلة نقرات لا يمكن إعادة تشغيلها ولا تسليمها
- **لا سجل تدقيق:** ضغطة زر طائشة واحدة، ولا أثر للقيمة الأصلية

عاقبة واقعية

تشرين الأول 2020: قرابة 16,000 حالة COVID-19 سقطت من التبليغ الوطني في إنكلترا — بسبب حد صفوف في صيغة Excel قديمة. السكربت لا يحمل أيّاً من أنماط الفشل هذه.

تشرح إطار البيانات

ثلاثة أجزاء وعقد واحد

- **الفهرس** — عناوين الصفوف: الصف الواحد مريض واحد أو عينة واحدة
- **الأعمدة** — المتغيرات السريرية، ولكل منها نوع بيانات واحد معلن
- **القيم** — القياسات نفسها، مخزنة في مصفوفات NumPy سريعة

السلسلة (Series)

`df["glucose"]` يستخرج عموداً واحداً: متجهاً معنونا لمتغير سريري واحد عبر جميع المرضى. ومعظم الإحصاءات أسئلة تُطرح على سلسلة.

- زوج أقواس واحد `df["glucose"]` يعيد سلسلة
- زوجاً أقواس `df[["glucose", "bmi"]]` يعيدان إطار بيانات أصغر

أنواع البيانات بيانات وصفية تشخيصية

أي نوع من القيم يسكن هذا العمود؟

النوع	ما يحتويه	مثال سريري
int64	أعداد صحيحة	العمر، عدد الصفحات
float64	أعداد عشرية وNaN	HbA1c، قيمة Ct
object	نصوص	patient_id، الجنس
bool	True / False	قناع الغلوكوز ≥ 126
datetime64	تواريخ	sample_date

لماذا يهم هذا

عمود غلوكوز مخزن نصاً (خلية واحدة فيها pending) يختفي بصمت من `describe()`. فحص أنواع البيانات هو قبول العينة.

بروتوكول النظرة الأولى: خمسة أوامر بلا معادلات

شغلها على كل ملف جديد، بهذا الترتيب

الأمـر	السؤال	النظير المختبري
<code>tail() / head()</code>	هل هذا الملف المتوقع؟	الفحص العياني
<code>shape</code>	كم مريضاً ومتغيراً؟	عدّ العينات
<code>info()</code>	الأنواع؟ المفقود؟	فحص القبول
<code>describe()</code>	التوزيعات؟	ملف ضبط الجودة
<code>value_counts()</code>	تكرارات الفئات؟	العد التفريقي

المئينات مجالات مرجعية

المجال المرجعي هو النسبة المركزية 95% من جمهرة سليمة (من المئين 2.5 إلى المئين 97.5).
و `describe()` يتحدث اللغة نفسها: صف 50% هو الوسيط — متين أمام القيمة الشاذة العبثية
الواحدة.

حَمَل الملف وانظر إليه

```
import pandas as pd
pd.set_option("display.width", 100)

df = pd.read_csv("diabetes_dataset.csv")
print(df.shape)
print(df[["glucose", "bmi", "age", "outcome"]].head(3))
```

```
(768, 9)
   glucose   bmi  age  outcome
0      134  41.3   31         0
1      149  34.7   37         0
2      106  31.5   46         1
```

يعيد shape دائماً (الصفوف، الأعمدة): 768 مريضاً وتسعة متغيرات — تحقق مقابل ما يدّعي نظام LIS أنه صدّره.

الصورة الشعاعية التقنية: info()

```
df.info()
```

```
RangeIndex: 768 entries, 0 to 767
#   Column      Non-Null Count  Dtype
---  -
0   pregnancies  768 non-null    int64
1   glucose      768 non-null    int64
5   bmi          768 non-null    float64
8   outcome      768 non-null    int64
dtypes: float64(2), int64(7)
memory usage: 54.1 KB
```

كل الأعمدة رقمية وكلها 768 غير فارغة: لا نتائج مفقودة ولا غلوكوز متكرر نصاً. والجدول كله يكلف 54 كيلوبايت.

الأمر describe(): تقرير مختبري للمجموعة كاملة

```
print(df[["glucose", "bmi", "age"]].describe().round(1).T)
```

	count	mean	std	min	25%	50%	75%	max
glucose	768.0	116.2	30.1	44.0	96.0	114.0	138.0	211.0
bmi	768.0	31.7	6.7	15.0	27.2	31.6	36.2	52.5
age	768.0	38.7	11.2	21.0	30.0	37.0	44.0	81.0

- المئين 75 للغلوكوز يساوي 138: ربع المرضى عند عتبة 126 ملغ/دل أو فوقها؛ والحد الأدنى 44 يستوجب التحقق من الملف السريري
- وسيط مؤشر كتلة الجسم 31.6: أكثر من نصف فئة الفحص هذه بدينة
- المتوسط يقارب الوسيط للغلوكوز: لا التواء جسيماً

معدل الانتشار بالأمر `value_counts()`

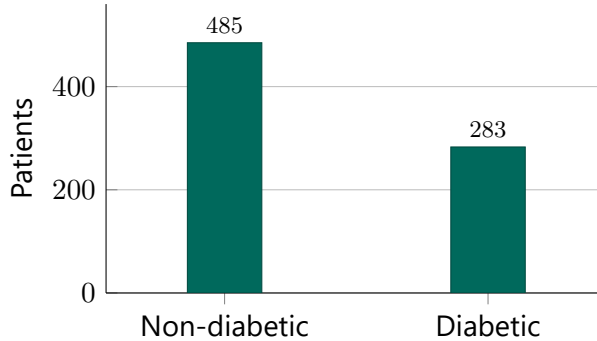
```
print(df["outcome"].value_counts())
print((df["outcome"].value_counts(normalize=True) * 100).round(1))
```

```
outcome
0    485
1    283
Name: count, dtype: int64

outcome
0    63.2
1    36.8
Name: proportion, dtype: float64
```

معدل انتشار 36.8% — في فئة فحص عالية الخطورة؛ وهذا السياق يجب أن يسافر مع الرقم إلى كل إحاطة.

توزيع المخرجات كما قيس فعلاً



485 مقابل 283 مريضاً — الأرقام التي أعادها `value_counts()` للتو.

ترتيب القيم المتطرفة وأول قناع منطقي

```
top5 = df.sort_values("glucose", ascending=False)
print(top5[["glucose", "bmi", "outcome"]].head(3))

high_glucose = df["glucose"] >= 126
print(high_glucose.sum())
print(round(high_glucose.mean() * 100, 1))
```

	glucose	bmi	outcome
510	211	52.5	1
206	207	48.0	1
659	205	46.6	1
292			
38.0			

True تُحسب 1: فالأمر `sum()` يعدّ المرضى ≥ 126 ملغ/دل، و `mean()` هو النسبة. الفلتر المنطقية الكاملة تصل في اليوم الرابع.

التحدي السريري: أحط المدير العام علماً

السيناريو

استخرجوا ملف `chemistry_panel.csv` (1200 مريض، 13 مادة مقيسة) للبرنامج الوطني للأمراض غير الانتقالية: الأدلة مع مذكرة من خمس جمل، في 30 دقيقة.

- 1 التحميل؛ الأبعاد؛ `describe().round(1).T;info()`
- 2 حددوا 3 متغيرات ذات حدود دنيا/قصوى مربية سريريا
- 3 الأقنعة: انتشار الغلوكوز ≥ 126 والسكر التراكمي $\geq 6.5\%$
- 4 رتبوا وفق مادة مربية واحدة؛ واعرضوا 5 مرضى متطرفين بمعرفاتهم
- 5 اكتبوا المذكرة: الاكتمال، والانتشار، والقيم المتطرفة، وإجراء واحد

التحليل	الحد الأدنى	الحد الأقصى	مصدر القلق
الغلوكوز (ملغ/دل)	55	186	هبوط سكر الدم
البوتاسيوم (ملي مول/لتر)	2.9	5.6	كلا الطرفين شاذ
الصوديوم (ملي مول/لتر)	130	152	من نقص إلى فرط الصوديوم

خلاصات اليوم

اليوم الثاني في ستة أسطر

- جداول البيانات تفشل بصمت؛ والسكربتات قابلة لإعادة الإنتاج والتدقيق
- إطار البيانات: فهرس وأعمدة وقيم؛ والعمود الواحد سلسلة
- خمسة أوامر (`head`, `shape`, `info`, `describe`, `value_counts`) تعطي الملف الكامل بصفر معادلات
- مئينات `describe()` مجالات مرجعية متكررة؛ والوسيط يتفوق على المتوسط في المتغيرات الملتوية
- العدّ بالقناع: `sum()` . للمرضى و `mean()` . للانتشار (الغلوكوز ≥ 126 : $38.0\% = 292/768$)
- البروتوكول نفسه استخرج ملفات السكري وصورة الدم والكيمياء دون تغيير

غداً — اليوم الثالث

messy_lab_data.csv: قيم مفقودة، قيم شاذة، مستحيلة، وصفوف وكيفية — تنظيف البيانات

اليوم 3

تنظيف البيانات الطبية

بيانات ملوثة في المدخلات، تشخيص خاطئ في المخرجات — إعداد بيانات مختبرية يمكنك الوثوق بها

90 دقيقة • 20 نظري — 40 تطبيق موجّه — 30 تحدّ سريري

اليوم 3 — جدول الأعمال والأهداف

دقائق اليوم التسعون

- 20 دقيقة — نظرية: لماذا تكون البيانات ملوثة، والعيوب الأربعة
- 40 دقيقة — تنظيف موجّه للملف `messy_lab_data.csv`
- 30 دقيقة — التحدي: اعتماد التصدير مع سجل التنظيف

ستكون قادراً على

- تدقيق فقدان؛ الحذف مقابل التعويض — وليس بصمت أبداً
- إزالة التكرارات التامة
- تحويل 9 رموز جنس إلى الترميز القياسي M/F
- حساب سياجي المدى الربيعي؛ والحكم على الشواذ سريراً
- تسليم ملف معتمد مع سجل تنظيف

لماذا تكون بيانات الرعاية الصحية ملوثة

مصانع التلوث الأربعة

- **النسخ اليدوي** — انزيادات الفاصلة العشرية: هيموغلوبين 9.9 يصبح 99؛ وغلوكوز 900 يصبح 9000
- **أعطال الأجهزة والواجهات** — إعادة المحاولات ترسل الدفعة نفسها مرتين
- **الخلط بين الوحدات** — ملغ/دل مقابل ملي مول/لتر (عامل 18 للغلوكوز)
- **دمج الأنظمة القديمة** — كل نظام يجلب قاموس ترميزه الخاص

الكلفة السريرية

النموذج المدرب على بيانات ملوثة يتعلم التلوث: قيمة غلوكوز واحدة مقدارها 9000 ملغ/دل ضاعفت متوسط أحد الأقضية وكادت تعيد توجيه ميزانية أنسولين.

البيانات المفقودة: حذف أم تعويض — قرار سريري

اكشف أولاً

`df.isna().sum()` ثم `df.isna().mean() * 100` (العبء النسبي المئوي)

قرّر سريرياً

- `patient_id` مفقود: احذف الصف — الهوية لا تُعوّض أبداً
- تحليلة مفقودة وسجل غني فيما عداها: عوّض **بالوسيط** (حصين ضد القيم الشاذة)

وليس بصمت أبداً

عمود راية وقيد في السجل لكل تعويض. ولا تعوّض أبداً الفراغات ذات الدلالة: قيمة `ct_value` في RT-PCR فارغة لأن النتيجة سلبية.

القيم الشاذة: نوعان وثلاث استراتيجيات

سياجا توکی للمدی الربيعی (طريقة حصينة)

والسياجان عند $Q_1 - 1.5 \times IQR$ و $Q_3 + 1.5 \times IQR$ ، مع Q_1, Q_3 من `quantile()`

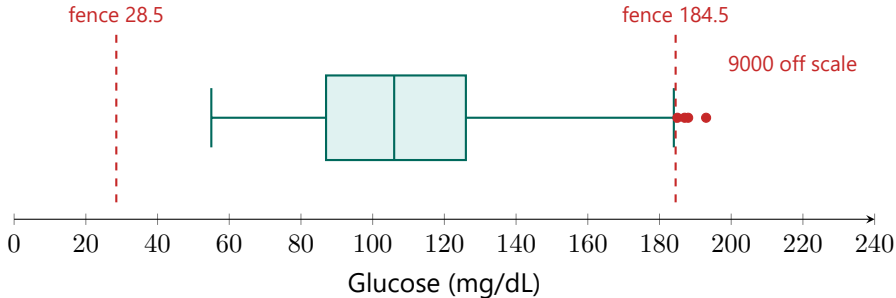
المستحيلة — أخطاء تسجيل

عمر 250، كرياتينين 2.0-، غلوكوز 9000 ملغ/دل، كريات بيض 900: احذف (وأبلغ المختبر المصدر)

المتطرفة لكنها حقيقية — المرضى الذين وُجدنا لأجلهم

غلوكوز 500 ملغ/دل في الحمض الكيتوني السكري؛ شحوم ثلاثية 900 ملغ/دل: **أبقِ وضع راية.**
القصّ يلفّق قيماً — ولا يكاد يُسوِّغ في الطب.

الغلوكوز في التصدير الملوث — قيم الجلسة الحقيقية



أربع قيم شاذة عند 185–193 ملغ/دل هي سكري غير مضبوط — أبقِ وضع راية. أما 9000 فلوحة مفاتيح لا مريض — احذف.

الخطوتان 1-2: التحميل وإحصاء الضرر

```
import pandas as pd
import numpy as np

lab = pd.read_csv("messy_lab_data.csv")
print(lab.shape)
print(lab.isna().sum())
```

```
(600, 8)
patient_id      0
age             52
glucose_mgdl    55
hgb             51
creatinine_mgdl 47
wbc             35
```

نسبة الفقدان بين 5.8 و 9.2 بالمئة لكل تحليلية؛ والمعرّفات كاملة.

الخطوتان 3-4: بوابة الهوية ثم التعويض بالوسيط

```
lab = lab.dropna(subset=["patient_id"])

glucose_median = lab["glucose_mgdl"].median()
print(glucose_median)

lab["glucose_imputed"] = lab["glucose_mgdl"].isna()
lab["glucose_mgdl"] = lab["glucose_mgdl"].fillna(glucose_median)
print(lab["glucose_imputed"].sum())
```

```
106.0
55
```

الوسيط 106 لا الوسط الحسابي 123.6 — فالوسط مسحوب بخطأ 9000. وعمود الرأية هو أثر التدقيق.

الخطوتان 5-6: التكرارات والترميز القياسي للجنس

```
print(lab.duplicated().sum())
lab = lab.drop_duplicates()

sex_clean = lab["sex"].str.strip().str.upper().str[0]
lab["sex"] = sex_clean.map({"M": "M", "F": "F"})
print(lab["sex"].value_counts())
```

```
20
sex
M    300
F    280
Name: count, dtype: int64
```

تسعة رموز صارت اثنين؛ والمجاميع محفوظة (580 = 300 + 280 صفًا). والرموز غير المطابقة كانت ستطفو بوصفها NaN — ولا تُخَمَّن أبداً.

الخطوة 7: سياجا المدى الربيعي وراية المراجعة

```
q1 = lab["glucose_mgdl"].quantile(0.25)
q3 = lab["glucose_mgdl"].quantile(0.75)
iqr = q3 - q1
lo = q1 - 1.5 * iqr
hi = q3 + 1.5 * iqr

is_outlier = ~lab["glucose_mgdl"].between(lo, hi)
print(lo, hi, is_outlier.sum())

flag = np.where(is_outlier, "review", "ok")
```

```
28.5 184.5 5
```

خمس قيم شاذة: 185 و187 و188 و193 (فرط سكر دم حقيقي — أبقيها) ثم 9000 (مستحيلة — الشريحة التالية تحذفها).

الخطوة 8: بوابات المعقولة الفسيولوجية

```

a_ok = lab["age"].between(0, 110) | lab["age"].isna()
g_ok = lab["glucose_mgdl"].between(20, 1000)
c_ok = (lab["creatinine_mgdl"].between(0.1, 20)
        | lab["creatinine_mgdl"].isna())
h_ok = lab["hgb"].between(3, 25) | lab["hgb"].isna()
w_ok = lab["wbc"].between(0.5, 100) | lab["wbc"].isna()

lab_clean = lab[a_ok & g_ok & c_ok & h_ok & w_ok].copy()
print(lab_clean.shape)

```

```
(575, 9)
```

شرط الإفلات `isna()` حاسم: بدونهُ يموت 169 صفّاً بدلاً من 5. والانحراف المعياري للغلوكوز يهبط من 382.1 إلى 28.7 ملغ/دل.

التحدي السريري — اعتماد التصدير (30 دقيقة)

السيناريو

انتقل مستشفى قضاء من نظام LIS عمره 15 سنة. تصديره يغذي ترصد السكري وتدريب نماذج الأسبوع القادم. ويشترط المدير العام مجموعة بيانات معتمدة مع سجل قابل للتدقيق.

مهامكم (ثنائيات)

- ① إحصاء القيم المفقودة — سَمّوا أسوأ عمود حرج
- ② بوابة الهوية، ثم تعويض الغلوكوز بالوسيط مع راية
- ③ عدّ التكرارات التامة وحذفها
- ④ الترميز القياسي M/F — أثبتوا حفظ المجاميع
- ⑤ سياجا المدى الربيعي وبوابات المعقولية — لا تحذفوا إلا المستحيل
- ⑥ السجل (المشكلة، العدد، الإجراء، المسوّغ)؛ ثم to_csv؛ ودافعوا سريرياً عن قراراتين

خلاصات اليوم — اليوم 3

- دقق قبل أن تصلح: `shape()` و `isna().sum()` و `value_counts()`
- لا تحذف إلا عند ضياع المعنى؛ وعوّض بالوسيط — مع راية دائماً، وفي السجل دائماً
- `duplicated()` ثم `drop_duplicates()`: كان 20 صفّاً مرضى معدودين مرتين
- مفهوم واحد، رمز واحد: `strip` ثم `upper` ثم `str[0]` ثم `map`
- السياجان يكشفان؛ والأطباء يقررون: احذف المستحيل، وضع راية على المتطرف لكنه حقيقي
- 600 صف خام صارت 575 صفّاً معتمداً — مع الدليل

غداً — اليوم 4

الفترة المنطقية المتقدمة وتحليلات العمليات السريرية: كل مدخن مصاب بالسكري تجاوز الستين في البصرة، بسطر موجه واحد.

اليوم 4

الفترة المنطقية المتقدمة وتحليلات العمليات السريرية

كل جملة شرطية لن تكتبها أبداً — الأقنعة المنطقية والتجميع ومؤشرات أداء المستشفى

90 دقيقة • 20 نظري — 40 تطبيق موجّه — 30 تحدّ سريري

اليوم 4 — جدول الأعمال والأهداف

دقائق اليوم التسعون

- 20 دقيقة — الأقنعة والتجميع والدمج والتواريخ
- 40 دقيقة — تطبيق موجّه: من ملفي CSV إلى إحاطة
- 30 دقيقة — تحدّي صفحة KPI الواحدة للمدير

ستكون قادراً على

- ترميز تعريف حالة بثلاثة معايير في سطر واحد
- دمج البيانات الديموغرافية مع النتائج المختبرية
- ترتيب الأقسام حسب معدل إعادة الرقود
- تسليم جدول KPI في صباح طلبه نفسه

القناع المنطقي: الأداة المركزية للدورة

مقارنة واحدة، عمود كامل

- `merged["age"] > 60` يُقِيم لجميع المرضى الـ 1,200 دفعة واحدة
- النتيجة: سلسلة من `False/True` — أي القناع
- `merged.loc[mask, cols]` يبقي صفوف `True` وحدها

القناع هو الإحصاء

- `mask.sum()` يعطي عدد الحالات (`True` تُحسب 1)
- `mask.mean()` يعطي الانتشار أو المعدل
- العتبة مكتوبة في الشيفرة: تعريفات حالات قابلة للتدقيق

الجمع بين المعايير — والقاعدة الواحدة غير القابلة للتفاوض

المفردات

- المعاملات: &(و) | (أو) ~ (نفي)
- مجموعات الفئات: `isin(["Basra", "Najaf", "Karbala"])`
- المجالات المرجعية: `hba1c.between(5.7, 6.4)`
- النص الحر: `department.str.contains("Medicine")`

الأقواس الإلزامية

- & أشد ارتباطاً من > و >=
- خطأ: `df["age"] > 60 & df["hba1c_pct"] >= 6.5`
- صواب: `(df["age"] > 60) & (df["hba1c_pct"] >= 6.5)`
- الكلمتان الإنكليزيتان or/and على الأعمدة: خطأ غموض قيمة الصحة

التقسيم-التطبيق-الجمع: تشبيه حامل الأنابيب

ثلاث حركات، سطر شيفرة واحد

- **التقسيم:** فرز 1,200 أنبوب في حوامل حسب القسم
- **التطبيق:** إجراء الإحصاء نفسه على كل حامل
- **الجمع:** سطر تقرير واحد لكل حامل

الصياغة في pandas

- `merged.groupby("department")["length_of_stay_days"].mean()`
- `agg(["mean", "median", "count"])` تعني ثلاثة إحصاءات في تمريرة واحدة
- `value_counts()` يرتب حسب التكرار؛ و `groupby(...).size()` يحافظ على ترتيب المجموعات
- المعدل هو متوسط عمود منطقي — لا حلقات ولا جمل شرطية

دمج الجداول: أي المرضى يبقى؟

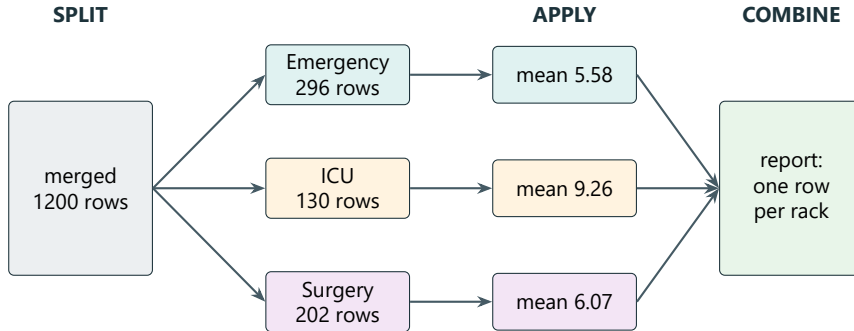
الدمج على المفتاح المشترك

- `patients.merge(chem, on="patient_id", how="inner")`
- **الداخلي:** فقط المرضى الموجودون في كلا الجدولين
- **الأيسر:** كل مريض راقد؛ وغير المفحوصين يحصلون على NaN

خيار سريري لا تقني

- تحليل الخطورة الكيميائي الحيوي: الداخلي (بيانات كاملة فقط)
- تدقيق تغطية الفحوصات: الأيسر (صفوف NaN هي النتيجة ذاتها)
- العادة الانعكاسية: `shape` . قبل كل دمج وبعده

التقسيم-التطبيق-الجمع، مرسوماً



قيم حقيقية: متوسط مدة الرقود لكل قسم (ثلاثة من ستة عوامل معروضة).

التطبيق الموجّه 1 — التحميل والدمج على patient_id

```
patients = pd.read_csv("patients_demographics.csv")
chem = pd.read_csv("chemistry_panel.csv")

merged = patients.merge(chem, on="patient_id", how="inner")
print("merged:", merged.shape)

cols = ["patient_id", "age", "department", "hba1c_pct"]
print(merged[cols].head(3))
```

```
merged: (1200, 21)
  patient_id  age  department  hba1c_pct
0  MOH-00001   53  Pediatrics        7.4
1  MOH-00002   23  Emergency        7.2
2  MOH-00003   63  Pediatrics        4.4
```

لم يُفقد أي صف: فلكل مريض راقد سجل كيميائي.

التطبيق الموجّه 2 — ثلاثة معايير في سطر مقنّع واحد

```
high_risk = (merged["age"] > 60) \
            & (merged["hba1c_pct"] >= 6.5) \
            & (merged["ldl_mgdl"] >= 160)

print("Case count:", high_risk.sum())
print(merged.loc[high_risk,
                 ["patient_id", "age", "hba1c_pct", "ldl_mgdl"]].head(3))
```

```
Case count: 14
   patient_id  age  hba1c_pct  ldl_mgdl
217  MOH-00218   75         7.0       178
278  MOH-00279   72         7.1       179
310  MOH-00311   72         7.0       164
```

أربعة عشر مريضاً مسقّى — قائمة إحالة لا مجرد رقم.

التطبيق الموجّه 3 — str.containsg betweeng isin

```
south = merged["governorate"].isin(["Basra", "Najaf", "Karbala"])
print("Southern admissions:", south.sum())

smoker_tg = (merged["smoker"] == "Yes") \
    & (merged["triglycerides_mgdl"] >= 200)
print("Smokers with high TG:", smoker_tg.sum())

prediab = merged["hba1c_pct"].between(5.7, 6.4)
print("Pre-diabetic band:", prediab.sum())
```

```
Southern admissions: 354
Smokers with high TG: 39
Pre-diabetic band: 302
```

التطبيق الموجّه 4 — agg و groupby والمعدل بوصفه متوسطاً

```

los_stats = (merged.groupby("department")["length_of_stay_days"]
              .agg(["mean", "median", "count"]).round(2))

merged["readmit"] = merged["readmitted_30d"] == "Yes"
readmit_rate = (merged.groupby("department")["readmit"]
                .mean().round(3)
                .sort_values(ascending=False))
print(readmit_rate.head(3))

```

department	
Oncology	0.161
Emergency	0.149
Surgery	0.129

يعيد قسم الأورام 16.1% من مرضاه خلال 30 يوماً — الرقم الذي يفتح مراجعة الملفات.

التطبيق الموجّه 5 — التواريخ والنمط الشهري

```
merged["admission_date"] = pd.to_datetime(merged["admission_date"])
merged["month"] = merged["admission_date"].dt.month

monthly = merged["month"].value_counts().sort_index()
print("Peak month:", monthly.idxmax(), "with", monthly.max())
print(monthly.tail(5))
```

```
Peak month: 8 with 131
month
8      131
9       88
10     100
11      92
12      99
```

آب أعلى بنحو الثلث من أهدأ الشهور.

التحدي السريري — صفحة مؤشرات الأداء الواحدة للمدير

طلب في الساعة 8:15 وموعده قبل اجتماع الظهيرة

على جدول الديموغرافيا والكيمياء المدمج:

- 1 الرقود لكل شهر؛ وتحديد شهر الذروة
- 2 متوسط ووسيط الرقود حسب الفئة العمرية (pd.cut: 0-17, 18-39, 40-64, 65+)
- 3 معدل إعادة الرقود لكل قسم، الأسوأ أولاً
- 4 المحافظات الثلاث الأولى بعدد الرقود
- 5 عدد مرضى الخطورة الحرجة: $\text{HbA1c} \geq 6.5\%$ وعمر ≥ 65
- 6 إطار بيانات KPI صغير واحد مع تفسير من خمس جمل

القواعد

لا حلقات ولا جمل شرطية — الأقنعة و `pd.cut` و `groupby` و `agg` و `value_counts` فقط.

اليوم 4 — الخلاصات الرئيسية

ما تستطيع فعله الآن دون حلقة واحدة

- الأقفنة: `sum()` . يعدّ الحالات و `mean()` . هو المعدل
- اجمع المعايير بـ `&` و `|` و `~` — أقواس حول كل مقارنة
- `str.contains / between / isin` للمجموعات والمديات والنصوص
- `groupby` مع `agg`: سير عمل حامل الأنابيب في سطر واحد
- `pivot_table` و `crosstab`: محاور إكسل قابلة لإعادة الإنتاج
- `merge`: الداخلي فوج كامل التوثيق، والأيسر تدقيق تغطية
- `pd.to_datetime` مع `.dt`: تتحول التواريخ إلى ترصد

غداً — اليوم 5

التمثيل البصري للبيانات: جداول KPI اليوم تصبح رسوماً، وتبدأ العلامات الحيوية برواية قصتها.

اليوم 5

التصوير البصري للبيانات وقصة العلامات الحيوية

من أعمدة الأرقام إلى قرارات تستطيع العين اتخاذها

90 دقيقة • 20 نظري — 40 تطبيق موجّه — 30 تحدّ سريري

اليوم 5 — جدول الأعمال والأهداف

اليوم في 90 دقيقة

مطابقة نوع المخطط مع السؤال السريري

- بناء أشكال موسومة بواسطة

`plt.subplots()`

رسم منحنى وبائي من الطوابع الزمنية الخام

- قراءة الخريطة الحرارية للارتباط قراءة سريرية

- التصدير بدقة 300 dpi لتقارير الوزارة

20 دقيقة — لماذا نصوّر البيانات؛ اختيار المخطط الصحيح

- 40 دقيقة — خمسة مخططات من ملفات CSV الخام للدورة

30 دقيقة — بناء لوحة ترصد المديرية

لماذا نصوّر البيانات؟ الشبكية أداة تشخيصية

الاكتشاف — ما تخفيه الجداول

- متوسطات متطابقة قد تخفي توزيعات متعاكسة (Anscombe)
- الشكل والالتواء والفجوات والقيم الشاذة لا تنجو إلا في الصورة
- القاعدة: لا تبْلُغ عن متوسط من دون رؤية توزيعه

التواصل — ما تقرؤه الإدارة

- مدرج تكراري بخط أحمر عند 126 ملغ/دل يكسب الموازنة
- الشكل الواحد يعبر الحدود المهنية؛ والجدول لا يفعل

المحاور الأمنية

محاور العدّ تبدأ من الصفر؛ وأي تكبير يُصرّح به في التسمية التوضيحية.

دليل اختيار المخطط

السؤال السريري	المخطط	اليوم
توزيع محلّ واحد؟	المدرج التكراري	الهيموغلوبين وقيم Ct
اختلاف المجموعات؟ شاذة؟	قيم المخطط الصندوقي	الغلوكوز حسب النتيجة
التغير عبر الزمن؟	المخطط الخطي	الإيجابيات الأسبوعية
ارتباط محلّين؟	مخطط الانتشار	كتلة الجسم مقابل الغلوكوز
متغيرات كثيرة معاً؟	الخريطة الحرارية	لوحة الكيمياء السريرية

القاعدة

السؤال هو الذي يختار المخطط — لا العادة أبداً ولا الزخرفة.

تشرح الشكل وجودة النشر

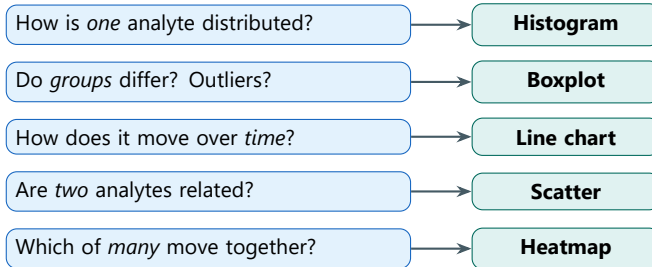
تشرح matplotlib

- Figure = اللوحة القماشية؛ Axes = نظام إحداثيات واحد
- `fig, ax = plt.subplots()` — مقبضان صريحان
- الرسم: `ax.plot` و `ax.scatter` و `ax.boxplot` و `ax.hist`
- قيم العتبة: `ax.axhline` و `ax.axvline`
- ترسم seaborn الإحصاء على كائن Axes نفسه (`ax=ax`)

قائمة تدقيق الوزارة

تسميات محاور مع الوحدات + عنوان + مفتاح رموز + `fig.savefig` مع `dpi=300`

خريطة القرار: من السؤال إلى المخطط



أول مدرج تكراري — الدمج ثم الرسم

```
cbc = pd.read_csv("cbc_results.csv")
demo = pd.read_csv("patients_demographics.csv")
blood = cbc.merge(demo[["patient_id", "age", "sex"]],
                  on="patient_id")
print(blood.shape)

fig, ax = plt.subplots(figsize=(8, 5))
ax.hist(blood["hgb"], bins=24, color="teal", edgecolor="white")
ax.set_xlabel("Hemoglobin (g/dL)")
ax.set_ylabel("Number of patients")
fig.savefig("hgb_hist.png", dpi=300)
```

```
(1200, 11)
```

اطبع الأبعاد دائماً بعد كل دمج — كي تلتقط انفجار الصفوف مبكراً.

مجموعتان سكانيتان — الهيموغلوبين حسب الجنس

```
male = blood[blood["sex"] == "M"]
female = blood[blood["sex"] == "F"]
ax.hist(male["hgb"], bins=24, alpha=0.6, label="Male")
ax.hist(female["hgb"], bins=24, alpha=0.6, label="Female")
ax.axvline(13.0, linestyle="--") # male lower limit
ax.axvline(12.0, linestyle="--") # female lower limit
ax.legend()
```

	count	mean	std
F	557	13.10	1.38
M	643	14.58	1.53

جرسان يفصل بينهما 1.5 غ/دل — لهذا يحمل الهيموغلوبين حدوداً خاصة بكل جنس. α يحوّل الطلاب الفوقي إلى مقارنة.

المخطط الصندوقي — الغلوكوز حسب نتيجة السكري

```
dm = pd.read_csv("diabetes_dataset.csv")
no_dm = dm[dm["outcome"] == 0]["glucose"]
yes_dm = dm[dm["outcome"] == 1]["glucose"]

fig, ax = plt.subplots(figsize=(7, 5))
ax.boxplot([no_dm, yes_dm],
            tick_labels=["No diabetes", "Diabetes"])
ax.axhline(126, linestyle="--", color="firebrick")
ax.set_ylabel("Plasma glucose (mg/dL)")
```

outcome	count	25%	50%	75%
0	485.0	88.0	104.0	121.0
1	283.0	120.0	136.0	151.0

الوسيطان 104 مقابل 136 — لكن الصندوقين يتداخلان: لا قيمة واحدة تشخص السكري.

المنحنى الوبائي في سطرين

```

pcr = pd.read_csv("rtPCR_results.csv",
                  parse_dates=["collection_date"])
pos = pcr[pcr["result"] == "Positive"]

week = pos["collection_date"].dt.isocalendar().week
weekly = pos.groupby(week)["sample_id"].count()

fig, ax = plt.subplots(figsize=(9, 4.5))
ax.plot(weekly.index, weekly.values, marker="o")

```

Weeks: 52 | peak week: 3 | cases: 12

مأزق

أسابيع ISO تُستأنف عند رأس السنة: 29–31 كانون الأول 2025 يقع في الأسبوع 1 من 2026. عبر

حدود السنوات، دقة حسب dt.to_period من ذلك

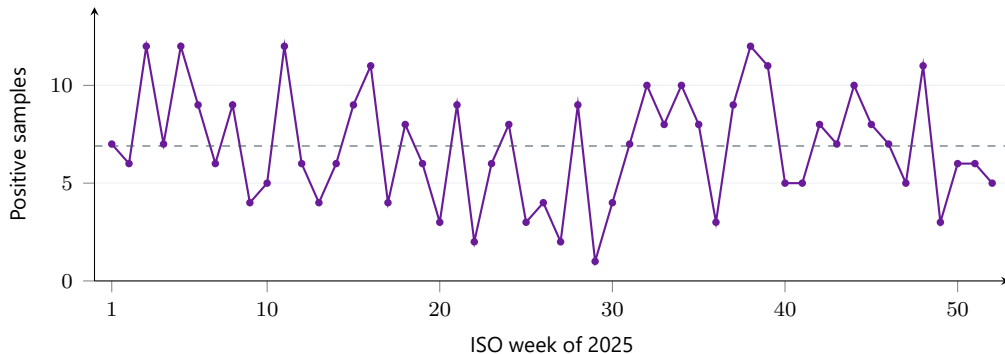
الخريطة الحرارية — لوحة الكيمياء في لمحة

```
chem = pd.read_csv("chemistry_panel.csv")
cols = ["glucose_mgdl", "hba1c_pct", "creatinine_mgdl",
        "urea_mgdl", "total_chol_mgdl", "ldl_mgdl"]
corr = chem[cols].corr()

sns.heatmap(corr, annot=True, fmt=".2f", cmap="vlag",
            vmin=-1, vmax=1, ax=ax)
fig.savefig("chem_heatmap.png", dpi=300)
```

- الغلوكوز مع HbA1c 0.85؛ الكرياتينين مع اليوريا 0.65؛ الكوليسترول مع LDL 0.90
- الكتل الدافئة هي فيزيولوجيا استُعيدت من البيانات
- خريطة حرارية تناقض الفيزيولوجيا تعني: دقق البيانات

المنحنى الوبائي لعام 2025 (بيانات الدورة الحقيقية)



متوسط 6.9 إيجابية في الأسبوع وذروة قدرها 12 — خط أساس متوطن. يصبح هذا النطاق عتبة الإنذار للموسم المقبل.

التحدي السريري — لوحة الترخيد

السيناريو

صفحة واحدة تُفتتح بها إجازات المدير العام يوم الاثنين: العبء الاستقلابي والوضع الوبائي.
الأداة: `plt.subplots(2, 2)`؛ الزمن: 30 دقيقة.

- 1 A: مدرج تكراري للغلوكوز + `axvline` أحمر عند 126 ملغ/دل
- 2 B: مخطط صندوقي للسكر التراكمي حسب الجنس + خط عند 6.5%
- 3 C: منحنى الإيجابيات الأسبوعية لفحص RT-PCR
- 4 D: انتشار العمر مقابل الغلوكوز مع `alpha`
- 5 `savefig tight_layout dpi 300` بدقة
- 6 اكتب النتائج الرئيسة الثلاث — الأرقام أولاً

الفحوص المطبوعة

نسبة الغلوكوز ≥ 126 ؛ وسيط السكر التراكمي لكل جنس؛ أسبوع الذروة؛ ارتباط العمر بالغلوكوز.

خلاصات اليوم

- خمسة أسئلة، خمسة مخططات: المدرج التكراري والصندوقي والخطي والانتشار والخريطة الحرارية — والسؤال هو الذي يختار.
- `fig, ax = plt.subplots()`؛ تسميات مع الوحدات وعنوان ومفتاح رموز و `dpi=300` — في كل مرة.
- المجالات المرجعية هي الـ 95% المركزية من توزيع أصحاء — و 5% من الطبيعيين يقعون خارجها بحكم التصميم.
- وجدنا اليوم: 15.9% من المرضى بغلوكوز ≥ 126 ؛ وسيط مرضى السكري 136 ملغ/دل؛ ومنحنى وبائي مستقر عند 6.9 في الأسبوع.
- المحاور الأمنية تحمي مصداقية المديرية — والنتائج السلبية (ارتباط العمر بالغلوكوز $r = 0.01$) هي نتائج أيضاً.

غداً — اليوم 6

هندسة الميزات (Feature Engineering): التدرجات التي رأتها عيناك اليوم تصبح المتغيرات التي

اليوم 6

هندسة الميزات للطب التنبؤي

تحويل أعمدة المختبر الخام إلى إشارات تستطيع الخوارزمية أن تتعلم منها

90 دقيقة • 20 نظري — 40 تطبيق موجّه — 30 تحدّ سريري

اليوم 6 — جدول الأعمال والأهداف

دقائق اليوم التسعون

- النظرية (20): الميزات، العلامات، الفئات، الترميز، درجات الخطورة
- التطبيق (40): هندسة مجموعة الكيمياء المدموجة
- التحدي (30): تهيئة heart_risk.csv لليوم السابع

ستصبح قادراً على

- اشتقاق النسب السريرية دون حلقات
- تعليم العتبات باستخدام np.where
- تقسيم المُحلَّلات إلى فئات باستخدام pd.cut
- ترميز الفئات باستخدام get_dummies / map
- بناء درجة خطورة مركبة قابلة للتدقيق

الميزات معرفة سريرية في هيئة أعمدة

مشكلة الترجمة

- النموذج لا يقرأ الطب — إنه يقرأ أعمدة رقمية
- الأعمدة الخام = قياسات؛ الميزات المهندسة = أحكام
- جودة الميزات تفوق عادةً في أهميتها اختيار الخوارزمية

أمثلة

- «HbA1c $\geq 6.5\%$ يعني السكري» ← عمود علامة 0/1
- «TC/HDL تتنبأ بالخطورة القلبية الوعائية» ← عمود نسبة
- «الردحات تختلف» ← عمود مؤشر واحد لكل قسم

لا خوارزمية تسترجع إشارة لم يحتوِها الجدول قط

الْعُدَّة الموجهة — لا حلقات ولا شروط

العملية	الأداة	مثال
نسبة مشتقة	العمليات الحسابية بين الأعمدة	TC/HDL
علامة ثنائية	<code>np.where</code>	الغلوكوز ≥ 126
فئات مرتبة	<code>pd.cut</code>	ثلاث فئات لـ HbA1c
ترميز ثنائي	<code>map + قاموس</code>	الجنس، التدخين
اسمي 0/1	<code>pd.get_dummies</code>	القسم
ميزات تواريخ	الموَصَّل <code>.dt</code>	شهر القبول
درجة مركبة	مجموع موزون	قائمة عمل الخطورة

تعبير واحد، كل المرضى الـ 1,200

كل أداة تعمل على العمود كاملاً دفعة واحدة — والنص البرمجي نفسه يتوسع دون تغيير إلى سجل وطني.

قاعدتا الترميز — وفخاهما

القاعدة 1: ثنائي / ترتيبى ← map

(M/F) sex، (Yes/No) smoker، ودرجات الشدة — عمود رقمى واحد أمين.

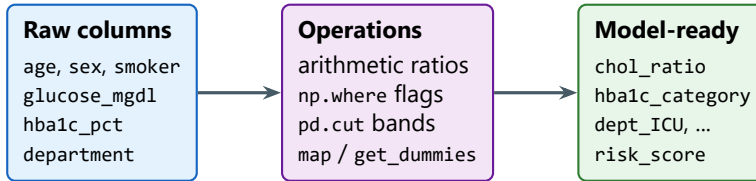
القاعدة 2: اسمى ← pd.get_dummies

القسم، والمحافطة، ونمط ألم الصدر: مؤشر 0/1 واحد لكل فئة — دون ترتيب زائف.

نمطا فشل صامتان

- تعيد map قيمة NaN للقيم غير المرمزة — تحقق دائماً من `isna().sum()` بعدها
- ترميز ICU=2، Emergency=1 يخبر النموذج الخطى بأن العناية المركزة ضعف الطوارئ — ميل موائم على خرافة

مسار هندسة الميزات



الكتلة اليسرى: ما يصدره المختبر. الكتلة اليمنى: ما تستهلكه نماذج اليوم السابع. والوسط هو حرفة اليوم.

العرض 1 — الدمج ثم اشتقاق نسبة TC/HDL

```
demo = pd.read_csv("patients_demographics.csv")
chem = pd.read_csv("chemistry_panel.csv")
df = demo.merge(chem, on="patient_id", how="inner")
print(df.shape)

df["chol_ratio"] = (df["total_chol_mgdl"]
                    / df["hdl_mgdl"]).round(2)
```

```
(1200, 21)
  patient_id  total_chol_mgdl  hdl_mgdl  chol_ratio
0  MOH-00001             161         45         3.58
1  MOH-00002             214         40         5.35
2  MOH-00003             234         44         5.32
```

● النسبة ≥ 5 تشير إلى خطورة قلبية وعائية مرتفعة؛ وأقصى المجموعة 11.10

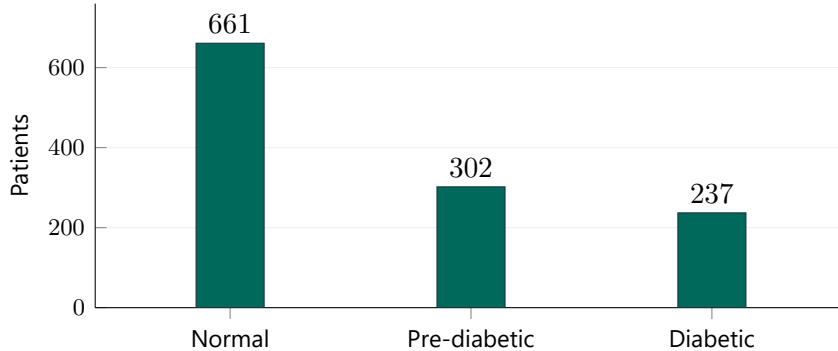
العرض 2 — العلامات وفئات ADA

```
df["hyperglycemia_flag"] = np.where(  
    df["glucose_mgdl"] >= 126, 1, 0)  
  
df["hba1c_category"] = pd.cut(  
    df["hba1c_pct"], bins=[0, 5.7, 6.5, 20],  
    labels=["Normal", "Pre-diabetic", "Diabetic"],  
    right=False)
```

```
hba1c_category  
Normal      661  
Pre-diabetic 302  
Diabetic    237
```

● `right=False`: الفترات مغلقة من اليسار، فتقع 6.5 في فئة السكري

فئات HbA1c في مجموعتنا



● مريض من كل خمسة يقع في فئة السكري — حجة لبرنامج تحرّ في رسم بياني واحد

العرض 3 — ترميز الفئات

```
df["sex_code"] = df["sex"].map({"M": 1, "F": 0})
df["smoker_code"] = df["smoker"].map({"Yes": 1, "No": 0})
print("Unmapped:", df["sex_code"].isna().sum())

dept = pd.get_dummies(df["department"],
                      prefix="dept", dtype=int)
dept.columns = dept.columns.str.replace(" ", "_")
df = pd.concat([df, dept], axis=1)
```

```
Unmapped: 0
dept_Emergency      296
dept_ICU             130
dept_Internal_Medicine  308
dept_Surgery        202
```

العرض 4 — درجة الخطورة الاستقلابية

```
df["risk_score"] = (
    2.0 * np.where(df["hba1c_pct"] >= 6.5, 1, 0)
    + 1.5 * df["smoker_code"]
    + 1.0 * np.where(df["age"] >= 65, 1, 0)
    + 1.0 * np.where(df["chol_ratio"] >= 5, 1, 0))

worklist = df.sort_values("risk_score", ascending=False)
cols = ["patient_id", "age", "hba1c_pct",
        "chol_ratio", "risk_score"]
print(worklist[cols].head(3).to_string(index=False))
```

patient_id	age	hba1c_pct	chol_ratio	risk_score
MOH-01054	95	8.1	7.11	5.5
MOH-00313	84	7.0	5.90	5.5
MOH-00339	71	6.9	5.77	5.5

● الدرجة المَرْتَبَة = قائمة عمل متابعة قابلة للاتصال بمعرّفات وطنية

الدرجات المركبة — الأداة الشفافة السابقة للذكاء الاصطناعي

لماذا نبنيها يدوياً أولاً

- قابلة للتدقيق: أي طبيب يستطيع إعادة حسابها
- قابلة للنشر هذا الأسبوع، دون بيانات تدريب
- سلف الانحدار اللوجستي — اليوم السابع يتعلم الأوزان

المعيار	الوزن
$\geq 6.5\% \text{ HbA1c}$	+2.0
مدخن حالياً	+1.5
≥ 65 العمر	+1.0
$\geq 5 \text{ TC/HDL}$	+1.0
الحد الأقصى	5.5

التحدي السريري — فرز طب القلب الوقائي

السيناريو

اهندس heart_risk.csv (900 إحالة لاختبار الجهد) ليصبح الجدول الرقمي الذي سيتدرب عليه اليوم السابع، وأنتج قائمة الفرز المؤقتة لليوم من أوزان اللجنة.

المهام

- 1 map للجنس وذبحة الجهد
- 2 get_dummies على نمط ألم الصدر
- 3 فئات عمرية باستخدام pd.cut
- 4 priority_score موزونة
- 5 فئات منخفضة / متوسطة / عالية
- 6 قائمة العشرة الأوائل؛ حفظ CSV

المهندس

أوزان اللجنة

- العمر ≥ 60 : +2
- ألم صدر ASY: +2
- انخفاض ST ≥ 2 ملم: +2
- الكوليسترول ≥ 240 : +1
- أقصى معدل قلب < 120 : +1
- علامة الغلوكوز الصائم: +1

خلاصات اليوم

- تتعلم النماذج من الأعمدة الرقمية — وهندسة الميزات ترمز المعرفة السريرية فيها
- np.where لعتبة واحدة؛ pd.cut مع right=False لفئات الأدلة الإرشادية (6.5 تقع في فئة السكري)
- map للثنائي/الترتيبي؛ pd.get_dummies للاسمي — لا تخرع ترتيباً أبداً
- بعد كل map: تحقق من isna().sum() — قيم NaN الصامتة تُفقد المرضى
- درجتنا القلبية الموزونة يدوياً رفعت انتشار المرض الحقيقي من 33% (منخفضة) إلى 73% (عالية) — قابلة للتدقيق ودون أي ذكاء اصطناعي
- heart_risk_engineered.csv جاهز للغد

غداً — اليوم السابع

الأسس الخوارزمية لتعلم الآلة: نموذج يتعلم الأوزان التي عيّناها اليوم بقرار لجنة.

اليوم 7

الأسس الخوارزمية لتعلم الآلة

التدريب والاختبار والانضباط المنهجي الذي يحفظ نزاهة الذكاء الاصطناعي التشخيصي

90 دقيقة • 20 نظري — 40 تطبيق موجّه — 30 تحدّ سريري

اليوم 7 — جدول الأعمال والأهداف

دقائق اليوم التسعون

- النظرية (20): التعلم الخاضع للإشراف، فرط التخصيص، المعايرة، التسرب
- التطبيق الموجّه (40): تحضير `diabetes_dataset.csv`
- التحدي (30): قسم المنهجية لملف `heart_risk.csv`

ستكون قادراً على

- بناء X و y ؛ والتقسيم مع `stratify=y`
- المعايرة دون تسرب؛ واستخدام `fit / predict / score`

ما هو تعلم الآلة

التعلم الخاضع للإشراف = استخلاص الأنماط من سجلات موسومة

- مصفوفة الميزات X : صف واحد لكل مريض، وعمود واحد لكل قياس — اللوحة المختبرية
- متجه الهدف y : التشخيص المؤكّد للمرضى أنفسهم، وبترتيب الصفوف نفسه
- تتعلم الخوارزمية $y \approx f(X)$ — ثم عليها أن تعمّم على مرضى لم ترهم قط

الترجمة السريرية

يتعلم الطبيب المبتدئ من حالات موثقة معلومة المآلات. وكذلك يفعل النموذج — على مستوى السجل الكامل.

الإقامة وامتحان البورد

- مجموعة التدريب (75%) = حالات الإقامة: للنموذج أن يدرسها بحرية
- مجموعة الاختبار (25%) = امتحان البورد: مرضى غير مرئيين، تُفتح مرة واحدة، في النهاية

فرط التخصيص – الخطر الأكبر منفرداً في الذكاء الاصطناعي الطبي

الطالب الذي حفظ امتحانات الأعوام الماضية: متقن في حالات التدريب، ضائع مع المرضى الجدد.
لا تكشفه إلا مجموعة اختبار محجوزة.

وسيطتان تجعلان التقسيم علمياً

- stratify=y: يحافظ كلا القسمين على معدل انتشار السجل
- random_state=42: تقسيم مطابق لكل مدقق

المعايرة وقاعدة منع التسرب

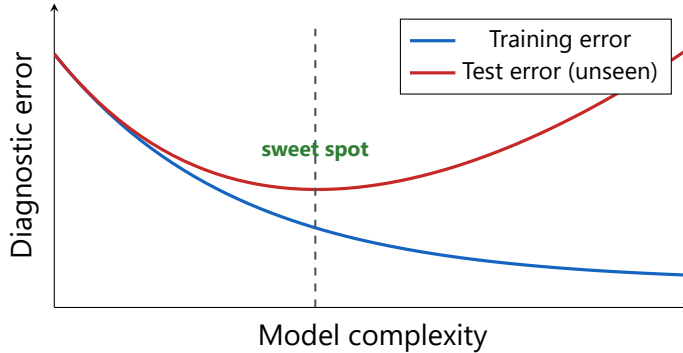
لماذا المعايرة القياسية؟ المقاييس تتفاوت بشدة (قيم لوحة حقيقية)

- متوسط الكوليسترول 185 ملغ/دل — ومتوسط البوتاسيوم 4.2 ملي مول/لتر
- من دون معايرة، تُسكّت الأرقام الكبيرة الأرقام الصغيرة
- StandardScaler: $z = (x - \bar{x})/s$ — الدرجة المعيارية لمخططات النمو وفحوص DXA

تسرب البيانات

المُعَاير يتعلم المتوسطات والانحرافات المعيارية. لائمه على السجل الكامل فيتسرب الامتحان إلى الإقامة. القاعدة: لائم على التدريب فقط، وحول المجموعتين.

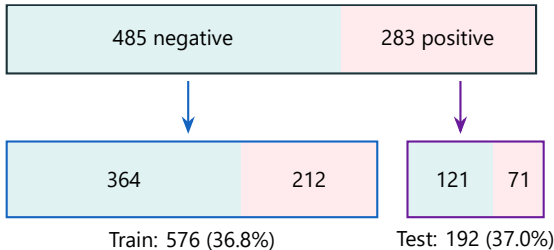
بصمة فرط التخصيص



- خطأ التدريب ينخفض باستمرار؛ والخطأ على المرضى غير المرئيين ينعطف عائداً
- المنحنى الأحمر وحده يقول الحقيقة عن مرضى المستقبل

التقسيم الطبقي — مزيج حالات محفوظ

Full registry: 768 patients (36.8% positive)



● stratify=y: للامتحان عبء المرض نفسه الذي للإقامة

الخطوتان 1-2: السجل والميزات والهدف

```
df = pd.read_csv("diabetes_dataset.csv")
print(df.shape)
print(df["outcome"].value_counts(normalize=True).round(3))

X = df.drop(columns="outcome")    # the lab panel
y = df["outcome"]                 # the confirmed diagnosis
print("X:", X.shape, " y:", y.shape)
```

```
(768, 9)
outcome
0    0.632
1    0.368
X: (768, 8)  y: (768,)
```

معدل الانتشار 36.8% — الرقم الذي سنطالب باسترداده من كلا القسمين.

الخطوتان 3-4: تقسيم طبقي قابل لإعادة الإنتاج

```

from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.25, stratify=y, random_state=42)

print("X_train:", X_train.shape, " X_test:", X_test.shape)
print(y_train.value_counts(normalize=True).round(3))
print(y_test.value_counts(normalize=True).round(3))

```

```

X_train: (576, 8)  X_test: (192, 8)
outcome
0      0.632
1      0.368
outcome
0      0.63
1      0.37

```

من دون stratify: تأرجح معدل انتشار الاختبار بين 32.8–41.1% عبر البذور.

الخطوة 5: المعايرة — الملازمة على التدريب فقط

```

from sklearn.preprocessing import StandardScaler

scaler = StandardScaler()
scaler.fit(X_train)          # learns mean, SD from train ONLY
X_train_s = pd.DataFrame(scaler.transform(X_train),
                          columns=X.columns, index=X_train.index)
X_test_s = pd.DataFrame(scaler.transform(X_test),
                         columns=X.columns, index=X_test.index)
print(X_train_s[["glucose", "bmi"]].agg(["mean", "std"]).round(3))

```

	glucose	bmi
mean	0.000	-0.000
std	1.001	1.001

يحطّ التدريب عند 0 / 1 بالضبط؛ أما جدول الاختبار المناظر فيحطّ قرب 0 / 1 فقط — البصمة المرئية لمُعَايير لم يرَ مجموعة الاختبار قط.

الخطوتان 6-7: مريض واحد، والقواعد

```
cols6 = ["glucose", "blood_pressure", "bmi", "age"]
print(X_train.loc[[711], cols6])           # raw units
print(X_train_s.loc[[711], cols6].round(2)) # z-scores

model = DummyClassifier(strategy="most_frequent")
model.fit(X_train_s, y_train)
print("baseline:", round(model.score(X_test_s, y_test), 3))
```

```
      glucose  blood_pressure  bmi  age
711      149             64  43.2  35
      glucose  blood_pressure  bmi  age
711      1.08             -0.45  1.72 -0.29
baseline: 0.63
```

التنبؤ بأن لا أحد مريض يسجل 63% أصلاً — أراضية الانتشار التي يجب على كل نموذج حقيقي أن يتجاوزها.

القواعد التي لا تتغير أبداً

كل أداة في scikit-learn، إلى الأبد

- 1 أنشئ المقدّر: `model = SomeEstimator(...)`
- 2 تعلم من الإقامة: `model.fit(X_train, y_train)`
- 3 طبّق على مرضى غير مرئيين: `model.predict(X_test)` — أو `transform` للمعالجات المسبقة

غداً

ضع `LogisticRegression` مكان `DummyClassifier` على السطر نفسه — ولا شيء آخر يتغير.

ملاءمة واحدة لكل دراسة

`fit` على التدريب فقط؛ و `predict/transform` في كل مكان آخر. `fit_transform(X)` قبل التقسيم = تسرب.

التحدي السريري — قسم منهجية الخطوة القلبية

السيناريو: بروتوكول دراسة وازاري، heart_risk.csv (900 مريض، 42.8% إيجابي)

- 1 حمل الملف، وأبلغ عن الأبعاد ومعدل انتشار target
- 2 ترميزات اليوم 6: اربط الجنس والذبة بـ map؛ وترميز أحادي لعمود chest_pain_type — من دون حلقات ولا جمل شرطية
- 3 ابن X (12 ميزة) و y؛ واذكر أسماء الميزات
- 4 قسّم 75/25 مع stratify=y و random_state=42
- 5 تحقق من توازن الفئات في كلا القسمين
- 6 عاير بـ StandardScaler ملائم على التدريب؛ وأثبت ذلك
- 7 اكتب فقرة المنهجية ذات الجمل الست؛ واطرح لماذا تُبطل المعايير قبل التقسيم الدراسة

خلاصات اليوم

- التعلم الخاضع للإشراف: ميزات X + هدف مؤكّد y ، بصفوف متحاذية
- مجموعة التدريب = الإقامة؛ ومجموعة الاختبار = امتحان البورد، تُفتح مرة واحدة
- فرط التخصيص = حفظ؛ ولا يُعتدّ إلا بالأداء على المرضى المحجوزين
- $stratify=y$ يحفظ معدل الانتشار؛ $random_state$ يجعل العلم قابلاً لإعادة الإنتاج
- $StandardScaler$ = الدرجة المعيارية السريية؛ لأتم على التدريب فقط
- التسرب يضخم الدقة بصمت — والجهات الرقابية تبحث عنه أولاً
- القواعد إلى الأبد: المقدّر، ثم fit ، ثم $predict/transform$

غداً — اليوم 8

تلتقي المصنّفات التشخيصية الحقيقية أقسامَ اليوم المحضرة بإتقان.

اليوم 8

بناء نماذج التصنيف التشخيصية

الانحدار اللوجستي والغابات العشوائية وآلات المتجهات الداعمة عند سرير المريض

90 دقيقة • 20 نظري — 40 تطبيق موجّه — 30 تحدّ سريري

اليوم 8 — جدول الأعمال والأهداف

دقائق اليوم التسعون

- 20 دقيقة — نظرية التصنيف: ثلاث عائلات من النماذج
- 40 دقيقة — تطبيق موجّه على `diabetes_dataset.csv`
- 30 دقيقة — التجربة الاسترشادية للمديرية على `heart_risk.csv`

ستكون قادراً على

- تدريب الانحدار اللوجستي والغابة العشوائية وآلة المتجهات الداعمة
- قراءة `predict_proba` بوصفها خطوة لكل مريض
- تدقيق المعاملات وأهميات الميزات سريراً
- شرح لماذا يجب ألا تُحسم قرارات النشر بالدقة وحدها

التصنيف: التنبؤ بفئة تشخيصية

ما الذي يفعله المصنّف

- يحوّل ميزات المريض (العمر، الغلوكوز، مؤشر كتلة الجسم) إلى فئة: 1 أو 0
- مخرجان: تصنيف قاطع واحتمال
- الاحتمال هو الأغنى سريراً بين الاثنين

الفرق عند سرير المريض

- «خطورة السكري: 82%» ⇒ اختبار تأكيدي وإرشاد صحي
- «التصنيف: 1» ⇒ لا شيء يوزن ولا شيء يُناقش

المبدأ

النموذج يدعم الطبيب — والطبيب صاحب القرار.

الانحدار اللوجستي: حسان العمل السريري

خطوتان

- مجموع موزون للميزات المعايّرة: $z = w_1x_1 + \dots + w_px_p + b$
- الدالة السينية تضغط z إلى احتمال: $\sigma(z) = 1/(1 + e^{-z})$

لماذا يثق به الأطباء

- كل معامل عامل خطورة اتجاهي
- الوزن الموجب يرفع الخطورة؛ والسالب يخفضها
- يمكن تدقيق النموذج كله سطرًا سطرًا — مثل درجة خطورة سريرية

تنبيه العتبة

الحد الفاصل الافتراضي 0.5 عُرف لا قانون — يُضبط في اليوم 9.

الغابة العشوائية: لجنة من الأشجار

كيف تعمل اللجنة

- نُنمّي 200 شجرة قرار، كل واحدة على **عينة تمهيد** من مرضى التدريب
- كل تفرع لا ينظر إلا في مجموعة جزئية عشوائية من الميزات
- التشخيص النهائي = **تصويت الأغلبية** لكل الأشجار

نقاط القوة

- الأنماط اللاخطية والتفاعلات؛ ترتيب `feature_importances_`

الأثمان

- لا أحد يستطيع قراءة 200 شجرة؛ والأهميات ترتب الاستخدام لا الاتجاه

آلة المتجهات الداعمة واختيار النموذج ودراسة حالة واقعية

آلة المتجهات الداعمة

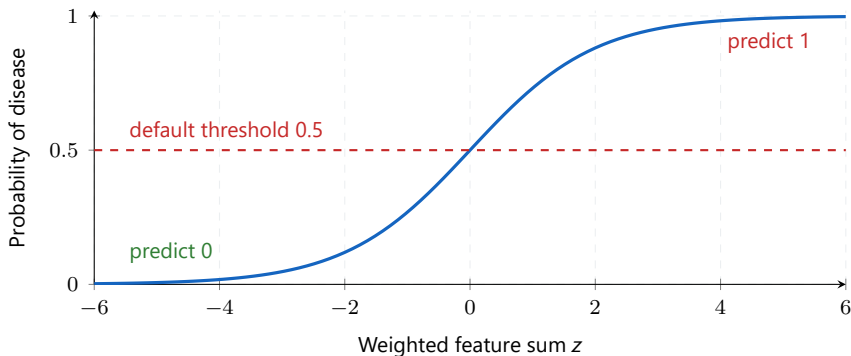
- تجد أوسع هامش أمان بين الفئتين
- المرضى الحديون (المتجهات الداعمة) يحددون الحد الفاصل
- تحتاج ميزات معيّنة؛ والاحتمالات فقط عبر $probability=True$

نموذج خط اليد للكشف المبكر عن الزهايمر (2024)

- مجموعة بيانات DARWIN: دقة 94.49%، حساسية 93.53%
- تحرّر غير باضع ومنخفض الكلفة — قلم وورقة
- لم يتحقق بعد: القيمة التنبؤية عند الانتشار المنخفض، والتحقق على الخط العربي، والاختبار الاستباقي

SVM	الغابة	اللوجستي	
منخفضة نعم	متوسطة نعم	عالية لا	قابلية التفسير الأنماط اللاخطية

الدالة السينية: من مجموع الميزات إلى الخطورة



- أي مجموع موزون يصبح خطورة بين 0 و 1 — قابلة للقراءة نسبة مئوية
- مريض عند 0.49 ومريض عند 0.02 يحصلان على التصنيف نفسه 0

التطبيق 1: مسار العمل وأولى التنبؤات

```
diab = pd.read_csv("diabetes_dataset.csv")
X = diab.drop(columns="outcome")
y = diab["outcome"]
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.25, stratify=y, random_state=42)
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)
log_model = LogisticRegression(max_iter=1000, random_state=42)
log_model.fit(X_train_scaled, y_train)
y_pred_log = log_model.predict(X_test_scaled)
```

```
First 10 predicted labels: [0 0 1 0 0 0 0 1 0 0]
First 10 true labels:    [0 0 1 0 1 0 1 1 0 0]
```

● المريض الخامس: تنبأنا بأنه سليم وهو سكري فعلاً — أول حالة فائتة

التطبيق 2: نسب الخطورة المئوية لكل مريض

```

risk = log_model.predict_proba(X_test_scaled)[: , 1]
review = pd.DataFrame({
    "glucose": X_test["glucose"].values,
    "bmi": X_test["bmi"].values, "age": X_test["age"].values,
    "risk_pct": (risk * 100).round(1),
    "predicted": y_pred_log,
    "true_outcome": y_test.values,
})
print(review.head(5).to_string(index=False))

```

glucose	bmi	age	risk_pct	predicted	true_outcome
108	33.5	38	29.5	0	0
151	48.2	65	96.8	1	1
89	31.5	45	17.5	0	1

● الصف الأخير: خطورة 17.5% لمریضة سكرية حقاً — حالة حدية فائتة تخفيها عتبة 0.5

التطبيق 3: تدقيق المعاملات

```
coef_table = pd.DataFrame({
    "feature": X.columns,
    "coefficient": log_model.coef_[0].round(3),
})
coef_table["direction"] = np.where(
    coef_table["coefficient"] > 0, "raises risk", "lowers risk")
coef_table = coef_table.sort_values("coefficient", ascending=False)
print(coef_table.to_string(index=False))
```

feature	coefficient	direction
glucose	1.099	raises risk
age	0.614	raises risk
bmi	0.596	raises risk
diabetes_pedigree	0.107	raises risk

● التدقيق ناجح: الغلوكوز والعمر ومؤشر كتلة الجسم — تماماً ما يتوقعه الطب

التطبيق 4: أهميات ميزات الغابة

```
rf_model = RandomForestClassifier(n_estimators=200,
                                  random_state=42)
rf_model.fit(X_train_scaled, y_train)
y_pred_rf = rf_model.predict(X_test_scaled)
importance = pd.DataFrame({
    "feature": X.columns,
    "importance": rf_model.feature_importances_.round(3),
}).sort_values("importance", ascending=False)
print(importance.head(4).to_string(index=False))
```

feature	importance
glucose	0.224
bmi	0.210
age	0.129
insulin	0.121

● الانسولين يظهر هنا: اللحنة ترى السنة الاخطية

د. حسام صلاح مهدي (وزارة الصحة)

بايثون والذكاء الاصطناعي للتشخيص الطبي

20 أغسطس 2026

140 / 109

التطبيق 5: لوحة النتائج — وفّذها

```
svm_model = SVC(kernel="rbf", probability=True, random_state=42)
svm_model.fit(X_train_scaled, y_train)
y_pred_svm = svm_model.predict(X_test_scaled)
scoreboard = pd.DataFrame({
    "model": ["Logistic regression", "Random forest", "SVM (RBF)"],
    "test_accuracy": [
        accuracy_score(y_test, y_pred_log),
        accuracy_score(y_test, y_pred_rf),
        accuracy_score(y_test, y_pred_svm),
    ],
}).round(3)
print(scoreboard.to_string(index=False))
```

model	test_accuracy
Logistic regression	0.802
Random forest	0.797
SVM (RBF)	0.771

الفخ في لوحة النتائج

الفخ

التنبؤ بعدم وجود سكري للجميع يحرز 0.63 — فالدقة هي أقل المقاييس السريرية إفادةً. واليوم 9 يستبدل بها غيرها.

التحدي السريري: التجربة الاسترشادية لأمراض القلب في المديرية

السيناريو

جربوا استرشادياً أداة دعم قرار تؤثر على مرض الشرايين التاجية المحتمل بين إحالات ألم الصدر، باستخدام heart_risk.csv — مع إعادة اشتقاق الميزات كما في اليوم 7.

- 1 الهندسة: map . للجنس والذبحة، و pd.get_dummies لنوع ألم الصدر
- 2 تقسيم طبقي 75/25؛ المعايير على مرضى التدريب وحدهم
- 3 تدريب النماذج الثلاثة؛ لوحة نتائج دقة واحدة
- 4 أعلى خمس أهميات للغابة — هل تطابق طب القلبية؟
- 5 المرضى الثلاثة الأعلى خطورة لوجستية لقائمة المراجعة ذات الأولوية
- 6 مذكرة: أي نموذج للتجربة، مع ذكر معيار واحد لا تستطيع الدقة التقاطه

خلاصات اليوم

- الاحتمال الكامن خلف التصنيف هو الناتج السريري: 0.82 تعني خطورة متنبأ بها 82%
- الانحدار اللوجستي = مجموع موزون + دالة سينية؛ والمعاملات عوامل خطورة اتجاهية قابلة للتدقيق
- الغابة العشوائية = أشجار تمهيد + تصويت أغلبية؛ قوية على الإشارات اللاخطية وضعيفة الشفافية
- آلة المتجهات الداعمة = أوسع هامش أمان؛ تتطلب المعايير، والاحتمالات المعايير عند الطلب فقط
- النموذج الأكثر شفافية كان الأدق أيضاً — والدقة تخطط الحالات الفائزة بالإنذارات الكاذبة: لا تنشر أبداً بالدقة وحدها
- النموذج يدعم؛ والطبيب يقرر

غداً — اليوم 9

الحساسية والنوعية ومنحنيات ROC واختيار عتبة تناسب كلفة التشخيص الفائت.

اليوم 9

التقييم النقدي للأداء التشخيصي

الحساسية والنوعية ومنحنى ROC والمساحة تحت المنحنى AUC — قمة الدورة

90 دقيقة • 20 نظري — 40 تطبيق موجب^{٤١} — 30 تحد^{١٤} سريري

اليوم 9 — جدول الأعمال والأهداف

دقائق اليوم التسعون

- 20 — فخ الدقة، مصفوفة الارتباك، المقاييس، معدل الانتشار، ROC وAUC
- 40 — تدقيق كامل لنموذج سكري اليوم الثامن
- 30 — ملف المسح القلبي المرفوع إلى المدير العام

ستكون قادراً على

- حساب الحساسية والنوعية وPPV وNPV يدوياً
- قراءة منحنيات ROC ورسمها ومقارنة قيم AUC
- تحريك عتبة والدفاع عن المقايضة
- كتابة مذكرة تقييم من خمس جمل

قاعدة اليوم

النموذج اختبار تشخيصي. قيمه كما يُقَيَّم أي اختبار — وأعد حساب كل رقم بنفسك.

فخ الدقة

دقة 98 بالمئة، وفائدة 100 بالمئة معدومة

انتشار المرض 2 بالمئة. نموذج يعلن أن الجميع أصحاء يصيب 9,800 مرة من كل 10,000: الدقة 0.98، والحساسية 0.00، والمرضى المكتشفون: لا أحد.

- الدقة تكافئ التوافق مع الفئة الأغلبية
- في المسح، الأغلبية أصحاء — فالدقة تخفي السوالب الكاذبة FN
- قاعدة المديرية: لا تبلغ عن الدقة وحدها أبداً؛ أضف دائماً الحساسية والنوعية وPPV وNPV ومعدل الانتشار

مصفوفة الارتباك بلغة سريرية

النموذج: سليم	النموذج: مريض
الحقيقة: سليم الحقيقة: مريض	FP — إنذار كاذب TP — التُّقط في الوقت المناسب

- الحساسية $TP / (TP + FN)$ =: من كل المرضى الحقيقيين، كم التقطنا؟
- النوعية $TN / (TN + FP)$ =: من كل الأصحاء، كم برّأنا؟
- $PPV = TP / (TP + FP)$ و $NPV = TN / (TN + FN)$ =: هل أثق بهذه النتيجة؟

اللاتناظر

FP يكلف فحصاً تأكيدياً وقلقاً. أما FN فيرسل مريضاً إلى بيته ومعه المرض — الخطأ الأخطر في الطب.

معدل الانتشار: انهيار القيمة التنبؤية الإيجابية

الاختبار نفسه — حساسية 0.90 ونوعية 0.90 — في مجموعتين سكانييتين من 10,000:

الإحالة (20%)	المسح (2%)	
1800 / 200	180 / 20	TP / FN
7200 / 800	8820 / 980	TN / FP
1800/2600 = 69.2%	180/1160 = 15.5%	PPV
97.3%	99.8%	NPV

- الحساسية والنوعية ترتحلان مع الاختبار؛ أما PPV و NPV فتخصان المجموعة السكانية
- نسبة الأرجحية 9 = $0.90/0.10 = LR+$: اضرب أرجحية ما قبل الاختبار في 9 تحصل على أرجحية ما بعده
- النموذج المعتمد في مستشفى يفقد PPV يوم ينتقل إلى المسح

ROC و AUC واختيار نقطة التشغيل

الاختبار الثنائي نقطة ROC واحدة.
الاستيفاء الخطي يطبع
 $AUC = (Sens + Spec)/2$ — وقد
يخس الاختبار. لا تقرأ AUC على عماها:
الانحياز للأتمتة خطأ تشخيصي.

- امسح العتبة؛ وارسم TPR مقابل FPR
- الركن المثالي (0, 1)؛ والقطر رمية عملة
- AUC هي احتمال ترتيب مريض عشوائي فوق سليم عشوائي
- 0.5 عديم الفائدة؛ 0.7–0.8 مقبول؛ 0.8–0.9 ممتاز؛ $0.9 >$ استثنائي — والسياق هو الحكم
- المسح → الحساسية أولاً؛ والتأكيد قبل الجراحة → النوعية أولاً
- مؤشر يودن $J = Sens + Spec - 1$. الافتراض المتوازن

العرض 1 — مصفوفة الارتباك والمقاييس يدوياً

```
proba = logit.predict_proba(X_test_s)[: , 1]
y_pred = (proba >= 0.50).astype(int) # the hidden default, made visible

tn, fp, fn, tp = confusion_matrix(y_test, y_pred).ravel()
sensitivity = tp / (tp + fn)
specificity = tn / (tn + fp)
ppv = tp / (tp + fp)
npv = tn / (tn + fn)
```

```
TN = 106 | FP = 15 | FN = 23 | TP = 48
Sensitivity : 0.676 (48 of 71 diabetics caught)
Specificity : 0.876 (106 of 121 healthy cleared)
PPV          : 0.762   NPV : 0.822   Accuracy : 0.802
```

دقة 80 بالمئة — ومع ذلك يخرج 23 من 71 مريض سكري دون كشف.

العرض 2 — منحنى ROC والمساحة AUC لنموذجين

```
from sklearn.metrics import roc_curve, auc

fpr, tpr, thresholds = roc_curve(y_test, proba)
auc_logit = auc(fpr, tpr)

forest = RandomForestClassifier(n_estimators=200, random_state=42)
forest.fit(X_train, y_train)          # trees need no scaling
proba_rf = forest.predict_proba(X_test)[: , 1]
fpr_rf, tpr_rf, thr_rf = roc_curve(y_test, proba_rf)
```

```
Thresholds swept by roc_curve: 58
AUC, logistic regression : 0.854
AUC, random forest      : 0.833
```

يفوز النموذج الأبسط القابل للتفسير — نتيجة تستحق التذوق.

العرض 3 — تحريك العتبة: رافعة المسح

```

y_pred_030 = (proba >= 0.30).astype(int)
tn3, fp3, fn3, tp3 = confusion_matrix(y_test, y_pred_030).ravel()

compare = pd.DataFrame(
    {"thr_0.50": [sensitivity, specificity, ppv, npv],
     "thr_0.30": [tp3/(tp3+fn3), tn3/(tn3+fp3),
                  tp3/(tp3+fp3), tn3/(tn3+fn3)]},
    index=["Sensitivity", "Specificity", "PPV", "NPV"]).round(3)

```

	thr_0.50	thr_0.30
Sensitivity	0.676	0.803
Specificity	0.876	0.711
PPV	0.762	0.620
NPV	0.822	0.860

مرضى السكري المفوتون: من 23 إلى 14. والثلث: الإنذارات الكاذبة من 15 إلى 35. لا حساسية مجانية — فقط مقايضة نتدكم بها

العرض 4 — المسح الموجّه ونقطة يودن

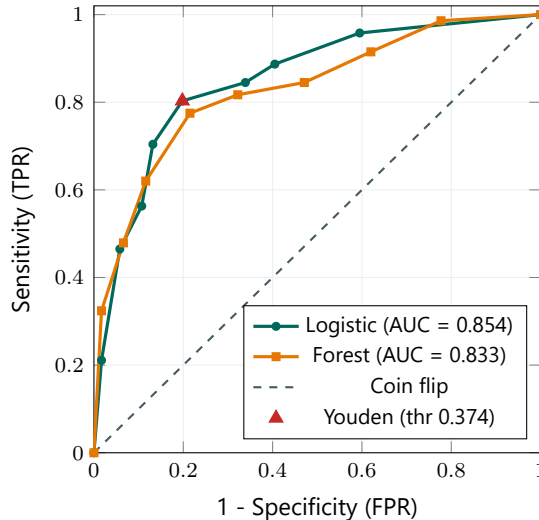
```
thr_grid = np.round(np.arange(0.10, 0.91, 0.10), 2)
pred_grid = (proba[:, None] >= thr_grid).astype(int) # 192 x 9 grid
truth = y_test.to_numpy()[:, None]
tp_g = ((pred_grid == 1) & (truth == 1)).sum(axis=0) # etc.

j = tpr - fpr # Youden J at every ROC threshold
best = np.argmax(j)
```

```
Youden J max : 0.604
Best threshold : 0.374
Sensitivity : 0.803
Specificity : 0.802
```

بُتّ واحد و `argmax` واحدة — بلا حلقات تكرارية، وجاهز للتدقيق خلال ثوانٍ.

منحنى ROC المقيس — مجموعة اختبار السكري (n = 192)



التحدي السريري — ملف المسح القلبي

السيناريو

نموذجان تجريبيان على `heart_risk.csv` مرشحان لبرنامج مسح قلبي وعائي وطني. المدير العام يقرر في الأسبوع المقبل. ابن ملف التقييم.

- 1 رقم، وقسّم 75/25 مع الحفاظ على نسب الفئات، وعاير للنموذج اللوجستي
- 2 درّب اللوجستي والغابة العشوائية؛ واستخرج الاحتمالات
- 3 مصفوفتا الارتباك عند 0.50؛ وتفكيك TN وFP وFN وTP
- 4 جدول المقاييس الكامل: الحساسية والنوعية وPPV وNPV والدقة وAUC
- 5 راكب منحنى ROC؛ واختر النموذج الذي يمضي قدماً
- 6 امسح العتبات؛ وأوصِ بنقطة تشغيل للمسح (كلف FN مقابل FP عند انتشار 10 بالمئة
- 7 الحالات المفوّتة لكل 1000 مفحوص + مذكرة من خمس جمل

خلاصات اليوم

- الدقة وحدها تضلل — بلّغ عن مجموعة المقاييس الكاملة مع معدل انتشارها
- FN هو الخطأ الأقل مغفرةً في الطب؛ ومصفوفة الارتباك هي الوثيقة الأولية
- تنهار PPV مع هبوط الانتشار (من 69% إلى 16%)؛ ولا تتغير الحساسية والنوعية
- العتبة رافعة سياسة: الانتقال من 0.50 إلى 0.30 خفض مرضى السكري المفوتين من 23 إلى 14، بثمن مرئي
- AUC تقارن مهارة الترتيب: 0.854 ممتاز، و0.791 درجة فرز
- المسح → الحساسية أولاً؛ والتأكيد → النوعية أولاً؛ ويودن الافتراض المتوازن
- أعد حساب كل مقياس بنفسك — الانحياز للأتمتة خطأ تشخيصي

غداً — اليوم 10

مسارات العمل والأتمتة والأخلاقيات — ومشروع التخرج: فريقك يسلم أداة تشخيصية كاملة ومقيّمة.

اليوم 10

مسارات العمل والأتمتة والأخلاقيات ومشروع التخرج

من دفتر الملاحظات إلى مسار عمل للمديرية — ومشروع التخرج

90 دقيقة • 20 نظري — 40 تطبيق موجّه — 30 تحدّ سريري

اليوم العاشر — جدول الأعمال والأهداف

مسار اليوم

- من التحليل إلى التشغيل
- Pipeline: المعاييرة + النموذج، كائن واحد
- تحقق متقاطع خماسي بقيمة AUC
- joblib: درّب مرة، وقمّ كل صباح
- الأخلاقيات: الخصوصية، والتحيز، والإنسان في حلقة القرار
- مشروع التخرج

ستكون قادراً على

- بناء مسار عمل محكم ضد التسرب
- إيراد AUC متوسطاً $\pm$ انحرافاً معيارياً
- الحفظ الدائم وإعادة التحميل والتقييم الدفعي
- تصدير قائمة عمل Excel مرتبة
- الدفاع عن خطة حوكمة

لماذا مسار العمل؟ ملف الطريقة المقفل

مقدّر واحد، بلا تسرب

- خطوات مسماة مرتبة: المعايرة أولاً، والنموذج آخرًا
- `pipe.fit` يعاير على بيانات التدريب فقط ثم يدرب
- `pipe.predict_proba` يعيد تطبيق المعايرة المخزونة

فشلان يصبحان مستحيلين

- **تسرب البيانات:** معايرة تسترق النظر إلى بيانات الاختبار فتضخم كل المقاييس — بصمت
- **نسيان الخطوات:** نموذج منشور يتلقى قيماً خاماً — مقايسة تعمل من دون محلولها الشاهد

المصادقة النزيفة والحفظ الدائم

التحقق المتقاطع الخماسي

- تقسيم واحد = سحبة يانصيب واحدة؛ خمس طيات = كل مريض يتحقق مرة واحدة بالضبط
- أورد المتوسط $\pm$ الانحراف المعياري لقيمة AUC والانحراف الواسع نتيجة بحد ذاته
- مسار العمل بأكمله يعاد ملائمة داخل كل طية — لا تسرب

joblib: النموذج دفعة كواشف

- `joblib.dump` يجمّد مسار العمل الملاءم في ملف واحد
- يخزن متوسطات المعايير + المعاملات + ترتيب الخطوات
- مرّقم الإصدار، ومؤرّخ، ولا يحزّر في مكانه أبداً

الأخلاقيات: الثوابت غير القابلة للتفاوض

ستة أركان لكل نموذج منشور في المديرية

- **الخصوصية:** احذف المعرّفات قبل التحليل؛ بيانات الدورة اصطناعية بالتصميم
- **الأطر القانونية:** HIPAA وGDPR وسياسة السرية في وزارة الصحة العراقية
- **التحيز الخوارزمي:** بغداد 336 مقابل كربلاء 91 مريضاً — نموذج بغدادي مع ضيوف
- **الانحياز للأتمتة:** يفرط البشر في الثقة بالآلات الوثيقة
- **الإنسان في حلقة القرار:** النموذج يقترح، والطبيب يتحقق ويوقع
- **المراقبة:** الجُمهرات والمقاييسات تنجرف — أعد التقييم كل ربع سنة

الطريق أمامنا — مع المحاذير

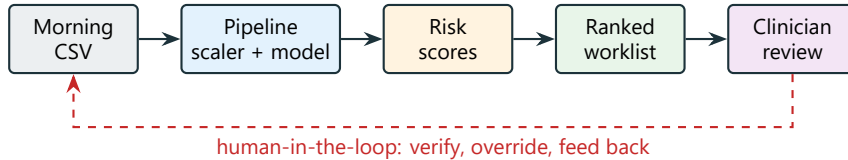
الحوسبة الكمومية

- محاكاة أصيلة للجزيئات: اكتشاف الأدوية، والجيโนมيات
- اليوم: أجهزة صغيرة كثيرة الضجيج — لا دور تشخيصياً روتينياً بعد

النماذج اللغوية الكبيرة

- عون حقيقي: التوثيق، والتلخيص، وهيكلة النص الحر
- حدود صارمة: الهلوسة، والتحيز المضخم
- بيانات المرضى لا تغادر البنية التحتية المؤسسية أبداً
- السياسة نفسها كسياسة اليوم: مسودة تخضع لتحقيق الطبيب

حلقة التشغيل



- كائن محفوظ واحد يطبق المعايير والتنبؤ تطبيقاً لا ينفصم
- كل راية تنتهي عند قرار بشري موَقَّع

بناء مسار العمل وإخضاعه للتحقق المتقاطع

```
df = pd.read_csv("diabetes_dataset.csv")
X = df.drop(columns=["outcome"])
y = df["outcome"]

pipe = Pipeline([
    ("scaler", StandardScaler()),
    ("model", LogisticRegression(max_iter=1000)),
])
auc_folds = cross_val_score(pipe, X, y, cv=5, scoring="roc_auc")
print("Fold AUCs:", auc_folds.round(3))
print("Mean AUC :", auc_folds.mean().round(3),
      "+/-", auc_folds.std().round(3))
```

```
Fold AUCs: [0.819 0.88  0.843 0.821 0.815]
Mean AUC : 0.836 +/- 0.025
```

درب مرة، واحتفظ به دائماً

```

X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, stratify=y, random_state=42)
pipe.fit(X_train, y_train)
joblib.dump(pipe, "diagnostic_model.joblib")

# --- a new session, a new analyst, the next morning ---
model = joblib.load("diagnostic_model.joblib")
proba_test = model.predict_proba(X_test)[: , 1]
print("Hold-out AUC of reloaded pipeline:",
      round(roc_auc_score(y_test, proba_test), 3))

```

```
Hold-out AUC of reloaded pipeline: 0.848
```

● حجز 80/20 بأسلوب الإنتاج لعرض النشر هذا — ومشروع التخرج يحافظ على بروتوكول 75/25 المألوف

● سكرتير الصباح لا يستدعى fit أبداً — تحميل وتقييم فقط

صباح الغد في عشرة أسطر

```
new_patients = pd.read_csv("new_patients.csv")
risk = model.predict_proba(new_patients)[: , 1]
worklist = pd.DataFrame({
    "case_id": "NEW-" + (new_patients.index + 1)
                        .astype(str).str.zfill(2),
    "glucose": new_patients["glucose"],
    "risk_pct": (risk * 100).round(1),
    "flag": np.where(risk >= 0.5, "REVIEW", "routine"),
}).sort_values("risk_pct", ascending=False)
worklist.to_excel("morning_worklist.xlsx", index=False)
```

```
case_id  glucose  risk_pct  flag
NEW-16    177      91.2  REVIEW
NEW-10    170      85.3  REVIEW
NEW-14    144      72.4  REVIEW
Exported morning_worklist.xlsx | 9 of 20 flagged
```

ملف مشروع التخرج التقييمي — العتبة خيار سريري

```
pred = (proba >= 0.35).astype(int)
tn, fp, fn, tp = confusion_matrix(y_test, pred).ravel()
```

```
heart_risk.csv | 5-fold CV AUC: 0.751 +/- 0.032 | test AUC: 0.791
At 0.50 -> TP 59 FP 22 FN 37 TN 107
Sens 0.615 Spec 0.829 PPV 0.728 NPV 0.743
At 0.35 -> TP 75 FP 47 FN 21 TN 82
Sens 0.781 Spec 0.636 PPV 0.615 NPV 0.796
```

- في الفرز: الحالة القلبية المفوتة هي الخطأ المكلف
- عتبة 0.35 تستعيد 16 حالة مفوتة لقاء 25 إيجابية كاذبة رخيصة
- معدل التعليم 54% — طاقة العيادة هي التي تقرر، لا دالة الخسارة

مشروع التخرج: فرز الخطورة القلبية

السيناريو

عيادة إحالات أمراض القلب تدير طابوراً واحداً غير مفاضل. ابن نموذج الفرز الاسترشادي على heart_risk.csv (900 مريض)، من البداية إلى النهاية.

- 1 التدقيق وتنظيف المعقولة بأقنعة between
- 2 التحليل الاستكشافي: متوسطات مجمعة + شكلان محفوظان
- 3 الميزات: أقنعة للثنائيات، و get_dummies لألم الصدر
- 4 التقسيم الطبقي المعياري 75/25؛ Pipeline؛ تحقق متقاطع خماسي لقيمة AUC
- 5 ملف الاختبار التقييمي: AUC، ومصفوفة الارتباك، والحساسية والنوعية وقيمتا التنبؤ
- 6 اختر عتبة فرز وبررها؛ ثم أعد الحساب
- 7 صَدّر قائمة العمل (to_excel) + joblib.dump
- 8 ملخص تنفيذي من عشرة أسطر إلى المدير العام

خلاصات اليوم — والتخرج

احملوا هذه معكم

- مسار العمل = مقدّر واحد؛ التسرب ونسيان الخطوات مستحيلان
- صادق بتحقيق خماسي متوسطاً $\pm$ انحرافاً معيارياً، لا بتقسيم واحد محظوظ
- joblib: النموذج أصل مؤسسي مرقّم الإصدار
- التقييم الدفعي: حقل، واقرأ، predict_proba، ورتّب، وصدّر
- أخف الهوية أولاً؛ ودقق التحيز؛ وراقب الانجراف فصلياً
- النموذج يقترح، والطبيب يتحقق ويوقع

انتهت الدورة

عشرة أيام، وعشر مهارات، ومديرية واحدة. قائمة العمل التالية التي ستعمل ستكون قائمتكم — تهانينا أيها الزملاء.

من البيانات إلى التشخيص

أصبحت الآن تنظفون البيانات الطبية وتحللونها وتصورونها وتمذجونها —
وتقيمون كل نموذج كما تقيمون أي فحص تشخيصي.

د. حسام صلاح مهدي • مديرية التشخيص — وزارة الصحة