XI

Kolika parametry nakreslíš slona?

Po této kapitole víš, že dokonalá shoda s minulostí nic nedokazuje — model se může nazpaměť naučit i šum. Počítá se jediné: jak se trefí do dat, která nikdy neviděl. A víš, jak to změřit.

Se čtyřmi parametry ti nakreslím slona, a s pěti mu rozhýbu chobot.

— John von Neumann, *„With four parameters I can fit an elephant, and with five I can make him wiggle his trunk.“ · připomínal Fermi; zaznamenal Freeman Dyson,* Nature *427 (2004)*

Slon se čtyřmi parametry

Na jaře 1953 přijel mladý Freeman Dyson do Chicaga za Enricem Fermim, tehdy nejváženějším experimentálním fyzikem světa. Vezl výsledky, na kterých jeho skupina dřela *dva roky*: pracné výpočty rozptylu mezonů na protonech podle takzvané pseudoskalární teorie. A ty výpočty krásně seděly na Fermiho vlastní data z chicagského cyklotronu. Dyson čekal uznání.

Fermi ho přijal vlídně a během pár minut mu ten dvouletý program zdvořile, ale nemilosrdně zboural. Namítl dvě věci. Za prvé: ta teorie nemá jasný fyzikální obraz ani pořádný matematický základ — je to jen recept na počítání. A pak se zeptal na to podstatné: „Kolik volných parametrů jste v tom výpočtu použili?“ Dyson odpověděl: „Čtyři.“ Fermi se usmál a řekl větu, kterou si Dyson zapamatoval na půl století: „Můj přítel Johnny von Neumann říkával: se čtyřmi parametry ti nafituju slona, a s pěti mu rozhýbu chobot.“

Tím byla rozprava u konce. Dyson odjel domů, teorii pustil k ledu — a udělal dobře: pseudoskalární teorie se ukázala jako slepá ulička. Krásná shoda s daty byla iluze koupená za čtyři čísla, která si člověk mohl nastavit skoro libovolně. Celý ten příběh Dyson sepsal až v roce 2004, ve vzpomínce pro *Nature* nazvané prostě *A*

batole — hlavní tok, čte každý

teenager — dílna, mechanismus

einstein — věž, matematika

Tři hloubky téže kapitoly: stejný pergamen, tři čtenáři.

meeting with Enrico Fermi. Napsal, že to byla jedna z nejcennějších lekcí jeho života.

Pramen: Freeman J. Dyson, „A meeting with Enrico Fermi“, Nature *427, 297 (22. 1. 2004). Výrok se připisuje von Neumannovi; Dyson ho zaznamenal z Fermiho úst. Skupina pracovala 1951–1953.*

Fermiho otázka — „kolik volných parametrů?“ — je detektor, který si v téhle kapitole osvojíš. Když ti někdo (nebo něco) ukáže model, který dokonale sedí na data, první, co tě má napadnout, není obdiv. Je to podezření: *kolik svobody jsi tomu modelu dal?* Protože dost svobody nakreslí slona. A jak uvidíš na konci kapitoly, dost svobody nakreslí slona *doopravdy*.

Papoušek a student

Vezmi si dva žáky před zkouškou. První je papoušek: naučil se nazpaměť odpovědi na všech dvě stě otázek z loňského testu. Slovo od slova, i s tečkami. Když mu položíš loňskou otázku, odpaluje odpověď bez zaváhání a bezchybně — vypadá jako génius. Druhý je student: loňské otázky si taky prošel, ale místo biflování pochopil, *proč* jsou odpovědi takové, jaké jsou. Na loňském testu udělá možná o chybu víc než papoušek; ztratí bod na detailu, který si papoušek pamatuje přesně.

A teď přijde letošní zkouška — s otázkami, které ani jeden z nich neviděl. Papoušek se zhroutí: jeho dokonalá paměť je k ničemu, protože otázky jsou nové a on umí jen přehrávat staré. Student projde, protože rozumí. **Rozdíl mezi papouškem a studentem nepoznáš na loňském testu — poznáš ho jedině na tom letošním, na otázkách, které nikdo z nich předem neviděl.**

přeučení (overfitting) — model se naučil trénovací data nazpaměť i s náhodným šumem; skvělý doma, mizerný venku. Papoušek.

nedoučení — opačná chudoba: model je tak prostý, že netrefí ani tvar dat. Přímka přes vlnu. O tom za chvíli.

Tohle je celá kapitola v jednom obrazu, tak si ho zapamatuj. Model, který se dat naučí nazpaměť — i s jejich náhodnými výkyvy, i se šumem —, se jmenuje **přeučený** (anglicky *overfitting*). Je to papoušek. Vypadá skvěle na datech, která viděl při učení, a selže na všem ostatním. Model, který pochopil tvar a šumu si nevšímá, je student. A protože „skvělý na loňském testu“ nic neznamená, potřebujeme letošní test: data, na kterých se model *neučil*.

To papouškování jsme ostatně viděli už v minulé kapitole a nechali jsme ho tam schválně nedopovězené. Vzpomeň si na tři tvary modelu nad cenami bytů: přímku, schodišťový strom a vlnovku k-nejbližších sousedů. Strom i vlnovka kopírovaly data ochotněji než přímka — a slíbil jsem, že tahle ochota kopírovat nemusí být ctnost. Teď ten červík dostává jméno: přehnaná ochota kopírovat je přeučení. Papouškování dat.

Klaudios Ptolemaios uměl staletí dopředu předpovídat polohy planet — a jeho model byl ontologicky *špatně: Země uprostřed, planety na kolech nasazených na kolech (epicyklech). Když predikce skřípala, přidal další epicyklus — další volný parametr. Ponaučení má dvě ostří: shoda s daty není důkaz pravdy (Ptolemaios se mýlil a trefoval se), ale ani shoda není bezcenná (staletí trefoval). Fit ≠ pravda; fit ≠ nula.*

Letošní test: rozděl data dřív, než začneš

Jak vyrobit „letošní zkoušku“, když máš jen jednu hromadu dat z minulosti? Jednoduše: část si schovej a nesahej na ni. Než začneš cokoli učit, náhodně oddělíš třeba pětinu příkladů stranou a zamkneš je do trezoru. Na zbytku model učíš — tomu se říká **trénovací data**. A schovanou pětinu — **testovací data** — vytáhneš jedinkrát, úplně nakonec, abys změřil, jak se model trefuje do příkladů, které při učení *neviděl*.

Testovací data jsou svatá. Kdo do nich během ladění nakoukne, kdo si podle nich vybere model a pak se jimi pochlubí, ten podvádí sám sebe — vrátil se papouškovat, jen o patro výš. Je to táž past jako pocit dřiny z kapitoly čtyři: měřit úspěch na tom, co už znám, je nejlevnější způsob, jak se obelhat.

TEENAGER · MECHANISMUS

Rozdělení dat je jeden řádek — a je to nejdůležitější řádek celého strojového učení:

```
trénovací_data, testovací_data = rozděl(data,
test = 20 %)

model.nauč_se(trénovací_data)
chyba_doma  = chyba(model, trénovací_data)   #
loňský test
chyba_venku = chyba(model, testovací_data)    #
letošní test
```

Zajímá tě `chyba_venku`. `chyba_doma` skoro vždycky vypadá líp — model si přece trénovací data osahal. Teprve rozdíl mezi oběma čísly ti řekne pravdu: když je `chyba_doma` maličká a `chyba_venku` velká, chytil ses papouška při činu. Model se naučil data, ne svět.

V praxi: `train_test_split` *ze scikit-learn, jeden řádek. Pozor na jednu věc — rozděluj* náhodně. *Když si na test schováš třeba jen nejdražší byty nebo poslední měsíc, neměříš neznámo, měříš svoje vlastní síto (ozvěna kap. 13: protokol vzniku vzorku rozhoduje).*

Jedna pětina stranou má ale vadu: co když sis do trezoru náhodou schoval zrovna těch pár nejzvláštnějších příkladů? Pak tě test oklame — buď moc přísně, nebo moc laskavě. Lék je otočit to dokola. Rozdělíš data na pět stejných dílů; pětkrát model naučíš, a pokaždé necháš jeden díl jako test a na zbylých čtyřech učíš. Každý díl si tak jednou zahraje na zkoušejícího. Pět čísel zprůměruješ — a máš odhad chyby, který nezávisí na jednom šťastném rozdělení. Říká se tomu **křížová validace**.

TEENAGER · MECHANISMUS

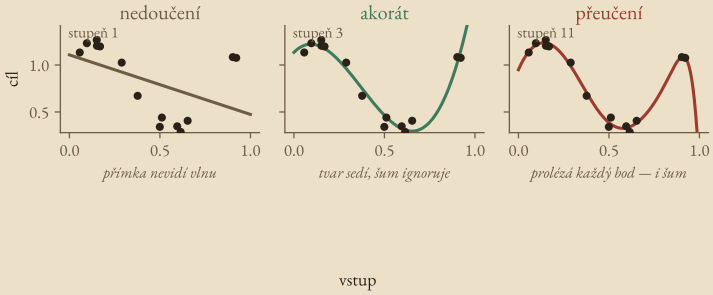
Křížová validace — každý kus dat si jednou zkusí být zkoušejícím:

```
křížová_validace(model, data, k = 5):
    rozděl data na k stejných dílů
    pro každý díl i:
        nauč model na všech dílech KROMĚ i
        změř chybu na dílu i        # ten teď
hraje test
    vrať průměr těch k chyb
```

`k = 5` je zvyk, ne zákon; `k = 10` je taky běžné. Cena je vyšší: natrénuješ model pětkrát místo jednou. Odměna: číslo, kterému se dá věřit, protože nestojí na jediném hodu kostkou při dělení dat.

Polynomiální hřiště: jak přeučení vypadá

Nejlíp to uvidíš na hračce, kterou si můžeš sám osahat. Vezmeme čtrnáct bodů, co leží kolem hladké vlny, a proložíme jimi křivku — polynom. Polynom má jeden knoflík složitosti: svůj **stupeň**. Stupeň 1 je přímka (dva parametry). Stupeň 3 je mírně zvlněná křivka (čtyři parametry). Stupeň 11 je divoká had (dvanáct parametrů), která se umí prohnout, kudy se jí zamane. Zvedej stupeň a dívej se, co to udělá:



Táž čtrnáctka bodů, tři stupně polynomu. Vlevo přímka nevidí vlnu — je líná, nedoučená*: netrefí ani tvar. Uprostřed stupeň 3 chytil tvar a šumu si nevšímá — to je student. Vpravo stupeň 11 prolézá každý bod, včetně náhodných výkyvů — papoušek. Který je „nejlepší"? Ten prostřední, ale to z tohohle obrázku ještě nepoznáš. Poznáš to až z dalšího.*

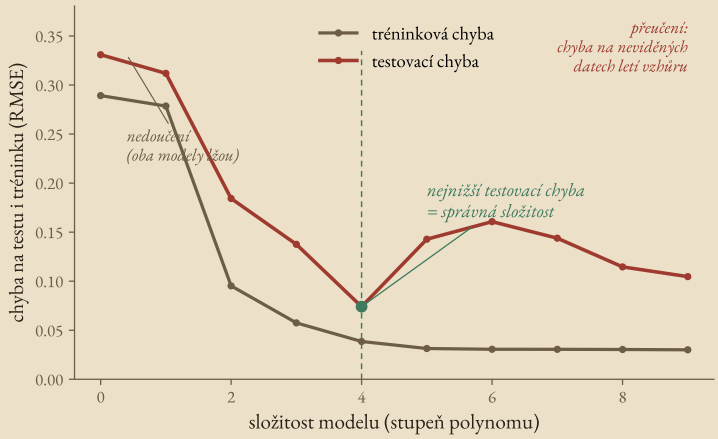
Vlevo je model *líný*. Přímka je tak prostá, že nedokáže vystihnout ani vlnu, kolem které body leží; mýlí se soustavně, nahoře i dole. Tomu říkáme **nedoučení** — model je tak chudý, že netrefí ani skutečný tvar. Vpravo je model *posedlý*. Stupeň 11 se prohýbá tak, aby prošel co nejblíž každému bodu — a tím věrně obkreslí i náhodný šum, který v datech nemá co dělat. Mezi body se přitom zmítá do nesmyslných výšek. To je přeučení v přímém přenosu.

A uprostřed je to, co chceme: model dost bohatý, aby chytil vlnu, a dost skromný, aby přeskočil šum. Jenže — a to je jádro celé kapitoly — **na tomhle obrázku nepoznáš, který je nejlepší**. Papoušek vpravo vypadá nejpřesněji: prochází body nejblíž! Kdybys měřil úspěch podle

shody s daty, která model viděl, vyhrál by ten nejpřeučenější. Musíš se zeptat jinak.

Křivky učení: kde se papoušek prozradí

Zeptáme se přesně tak, jak nás naučil papoušek se studentem: změříme chybu i na datech, která model *neviděl*. Vezmeme stejnou úlohu, ale teď pro každý stupeň polynomu spočítáme dvě čísla — chybu na trénovacích datech (loňský test) a chybu na čerstvých testovacích datech (letošní test) — a obě vyneseme proti složitosti modelu:



Hrdinský graf kapitoly — a diagnóza, kterou budeš číst po zbytek života. Šedá (trénink) klesá pořád: čím složitější model, tím líp našprtá to, co vidí — až k nule. Sanguine (test) je jiná: nejdřív klesá (model se opravdu učí tvar), dosáhne dna, a pak stoupá — model začal papouškovat šum a na neviděných datech to platí. To dno je správná složitost. Nalevo od něj nedoučení, napravo přeučení.

Přečti ten obrázek pomalu, protože je to nejcennější dovednost kapitoly. Šedá křivka — chyba na tréninku — klesá monotónně: čím víc parametrů modelu dáš, tím věrněji obkreslí data, která má před sebou. Sám o sobě ten pokles nic neznamená; je to jen měřítko toho, jak dobrý papoušek dokážeš vyrobit. Sanguine křivka — chyba na neviděných datech — vypráví skutečný příběh. Zprvu klesá souběžně: model se poctivě učí tvar. Pak narazí na dno. A od dna začne **stoupat**, i když šedá křivka pořád klesá. To je okamžik, kdy se model přestal učit svět a začal se učit šum: doma se lepší, venku se horší. Papoušek se prozradil.

To dno sanguine křivky je odpověď na Fermiho otázku. Nalevo od něj je model moc prostý (nedoučení — vysoké **vychýlení**: tvrdohlavě nevidí, co v datech je). Napravo je moc bohatý (přeučení — vysoký **rozptyl**: těká za každým výkyvem). Správná složitost sedí v tom údolí — a najdeš ji jedině tak, že měříš venku, ne doma.

Dvě jména pro dvě nemoci, budou se ti hodit ve věži. Vychýlený (biased) model je tvrdohlavý — nevidí, co v datech je (přímka přes vlnu). Rozptýlený (high-variance) model je těkavý — vidí i to, co tam není (had přes šum). Nedoučení = vychýlení; přeučení = rozptyl.

Regularizace: pokuta za složitost

Co s tím? Jedna možnost je hledat správný stupeň ručně, jak jsme to udělali teď. Druhá, elegantnější, nechá model složitý — ale **zdaní mu složitost**. Místo abys polynomu zakázal vysoké stupně, řekneš mu:

klidně je používej, ale za každý velký parametr zaplatíš pokutu. Model pak sám od sebe nechá nepotřebné parametry splasknout k nule, protože se mu nevyplatí je platit. Té pokutě za složitost se říká **regularizace**.

TEENAGER · MECHANISMUS

Učení bez pokuty hledá jen nejmenší chybu:

```
minimalizuj:  chyba(model, trénovací_data)
```

Regularizované učení přidá do rovnice cenu za mohutnost parametrů — jedno kolečko `síla_regularizace` říká, jak přísně:

```
minimalizuj:  chyba(model, data)  +
síla_regularizace · velikost(parametry)
```

Když je `síla_regularizace` nula, jsi zpátky u papouška. Když je obrovská, umlčíš i užitečné parametry a spadneš do nedoučení. Správnou sílu — jak jinak — vybereš křížovou validací: zkusíš několik hodnot a necháš rozhodnout chybu na datech, která model neviděl.

Dvě běžné pokuty se liší tím, jak *velikost parametrů měří. L2 (hřebenová, ridge) trestá součet čtverců — parametry hladce zmenší, ale k nule je nedotlačí. L1 (laso, lasso) trestá součet absolutních hodnot — nepotřebné parametry usekne rovnou na nulu, a tím sám vybere, na čem záleží. Proč zrovna tyhle dvě, uvidíš ve věži.*

> *Regularizace je pokuta za to, že model bere víc pozornosti, než potřebuje. Kdyby existovala pro schůze, výbory a e-mailové kopie, žili bychom v moudřejším světě. Bohužel funguje jen na parametry — ty totiž, na rozdíl od kolegů, splasknou k nule bez urážky.*

Věž: bias–variance, a proč zrovna L1 a L2

EINSTEIN · MATEMATIKA — přeskočitelné

Rozklad chyby na vychýlení a rozptyl. Proč vůbec existuje to údolí na sanguine křivce? Protože chyba na neviděných datech se skládá ze tří kusů, z nichž dva táhnou proti sobě. Předpokládej, že data vznikají jako $y = f(x) + \varepsilon$, kde f je pravda a ε je šum s rozptylem σ^2 a nulovým průměrem. Náš model $\hat{f}$ se učí z konkrétního (náhodného) trénovacího vzorku. Očekávaná čtvercová chyba v bodě x, přes všechny možné tréninkové vzorky, se rozpadne na:

$$\mathbb{E}\Big[\big(y - \hat{f}(x)\big)^2\Big] = \Big(\mathrm{Bias}\big[\hat{f}(x)\big]\Big)^2 + \mathrm{Var}\big[\hat{f}(x)\big] + \sigma^2$$

Odvození je pár řádků: rozepiš $y - \hat{f} = \big(f - \mathbb{E}\hat{f}\big) + \big(\mathbb{E}\hat{f} - \hat{f}\big) + \varepsilon$, umocni a vezmi střední hodnotu; smíšené členy vypadnou, protože ε má nulový průměr a je nezávislý a protože $\mathbb{E}\big[\mathbb{E}\hat{f} - \hat{f}\big] = 0$. Zůstanou přesně tři členy.

EINSTEIN · MATEMATIKA — přeskočitelné

Co ty tři členy znamenají. Čti je zprava:

- σ^2 je **neredukovatelný šum** — cena, kterou nezaplatí žádný model, protože pochází z dat samých. Podlaha, pod kterou se nedostaneš.
- $\mathrm{Bias}\left[\hat{f}\right] = \mathbb{E}\hat{f} - f$ je **vychýlení**: jak moc se model míjí s pravdou i v průměru přes všechny vzorky. Vysoké u prosté třídy modelů — přímka *nemůže* být vlna, ať ji učíš na čemkoli. To je nedoučení.
- $\mathrm{Var}\left[\hat{f}\right]$ je **rozptyl**: jak moc modelem cuká výměna trénovacího vzorku. Vysoký u bohaté třídy — polynom stupně 11 nakreslí pokaždé jinou divokou křivku, podle toho, kde zrovna padl šum. To je přeučení.

A tady je celý tah kapitoly v jedné větě: **zvětšuješ-li složitost modelu, vychýlení klesá, ale rozptyl roste.** Jejich součet má minimum — to je dno sanguine křivky. Přeučení není hloupost modelu; je to zvítězivší rozptyl.

EINSTEIN · MATEMATIKA — přeskočitelné

Regularizace jako předchozí přesvědčení (MAP). Proč zrovna součet čtverců (L2) nebo absolutních hodnot (L1) parametrů? Protože obojí je maximum aposteriorní pravděpodobnosti (MAP) s konkrétním *priorem* na parametry w. Bayesovsky: hledáme nejpravděpodobnější parametry *po* zhlédnutí dat,

$$\hat{w} = \arg\max_{w} \underbrace{\log P(\text{data} \mid w)}_{\text{věrohodnost}} + \underbrace{\log P(w)}_{\text{prior}}.$$

Věrohodnost pod gaussovským šumem je záporný součet čtverců chyb (viz věž kap. 10). Zbývá prior — naše přesvědčení o tom, jaké parametry jsou rozumné, *dřív* než uvidíme data:

- Gaussovský prior $w \sim \mathcal{N}(0, \tau^2)$ dá $\log P(w) \propto -\sum w_j^2$ — to je **L2**. Přesvědčení: „parametry jsou nejspíš malé.“
- Laplaceův prior $P(w) \propto e^{-|w|/b}$ dá $\log P(w) \propto -\sum |w_j|$ — to je **L1**. Jeho špičatý vrchol v nule vytlačí slabé parametry přesně na nulu — proto laso samo vybírá, na čem záleží.

Regularizace tedy není trik navíc: je to poctivé přiznání, že do modelu vcházíme s nějakým očekáváním, a síla pokuty je jen jazýček vah mezi tím, co říkají data, a tím, co jsme čekali předem.

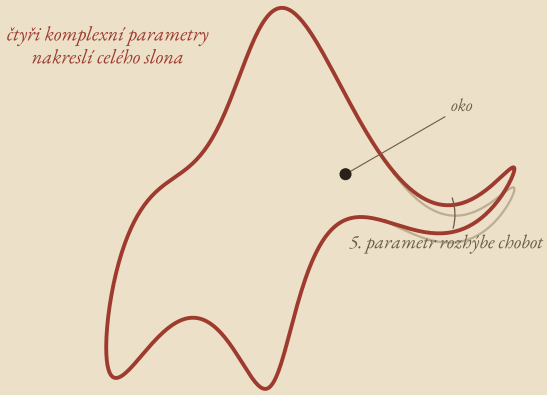
A teď doopravdy: slon ze čtyř parametrů

Von Neumannův výrok zní jako nadsázka. Není. V roce 2010 tři biofyzici — Mayer, Khairy a Howard — vzali čtyři **komplexní** čísla (každé

EINSTEIN

Double descent. U dnešních obřích sítí, které mají víc parametrů než dat, sanguine křivka za vrcholem přeučení znovu klesne *— někdy pod původní dno. Klasická křivka platí dál (je to speciální případ); jen se ukázalo, že mapa složitosti má ještě jeden svah, který se v roce von Neumanna nedal tušit. Lekce téhle kapitoly stojí; rozšířila se mapa, ne pravidlo. Kdo umí číst chybu na neviděných datech, projde obojím.*

nese dvě reálná, takže dohromady osm laditelných hodnot) a použili je jako koeficienty Fourierova rozvoje pro obrys tvaru. Výsledek:



věrná rekonstrukce dle Mayer, Khairy & Howard, Am. J. Phys. 78, 648 (2010)

Věrná rekonstrukce podle Mayer, Khairy & Howard, „Drawing an elephant with four complex parameters“, Am. J. Phys. *78, 648 (2010). Čtyři komplexní čísla nakreslí obrys i s nohama a hlavou; páté číslo je oko* a *amplituda, s níž se rozhýbe chobot — přesně jak von Neumann slíbil. Duch chobotu na pozadí je právě to vlnění pátým parametrem.*

To je pointa celé kapitoly, převedená z metafory na obrázek. Dost volných parametrů nakreslí cokoli — slona, tvá minulá data, šum na burze, loňské počasí. Shoda s tím, co model viděl, není důkaz porozumění; je to jen důkaz, že model měl dost svobody se přizpůsobit. Von Neumann to věděl u čtyř parametrů; Fermi tím za pět minut zboural dva roky Dysonovy práce; a ty to teď umíš změřit sám, na datech, která model neviděl.

A protože je to kniha o vibe codingu, dopověďme to natvrdo. Když ti dnes model — nebo asistent, kterému říkáš, ať „nafituje“ vztah v datech — ukáže křivku, co prochází všemi tvými body, neraduj se. Zeptej se Fermiho otázkou: kolik volných parametrů to mělo? A pak udělej to jediné, co rozhodne: změř chybu na datech, která to při učení nevidělo. Jinak si možná právě koupil nádherně nakresleného slona a spletl si ho s porozuměním.

DEMO

`lesk-a-bida-vibecodingu.gaussalgo.com/d/prefit`

Posuvník stupně polynomu vs. tréninková a testovací chyba — přefituj si sám a dívej se, jak sanguine křivka klesne a pak zase vyletí nahoru.

Nitě, které odtud vedou dál. Do minulé kapitoly: pamatuješ, jak čára měřila svou chybu jen na bytech, které viděla? Teď víš, proč to byla past — a jak ji obejít. Do kapitoly patnácté: velký jazykový model je pořád jen model s parametry a chybou, jen jich má miliardy — a otázka „naučil se jazyk, nebo si zapamatoval korpus?“ je přesně náš papoušek versus student, jen ve velkém. A do kapitoly páté: když se model učí z vlastních

výstupů kolo za kolem, je to přeučení celé civilizace na sebe samu — had, který požírá vlastní ocas a myslí si, že se nasytil.

„Jak bych poznal, že se pletu?“

Konkrétní test k této kapitole je zároveň nejkratší návyk, jaký si z ní můžeš odnést. Než uvěříš, že model něco *umí* — ať je to přímka přes ceny bytů, nebo asistent, který ti „vyřešil“ úlohu —, polož dvě otázky. První: *viděl ta data při tréninku?* Druhá, důležitější: *jaká je jeho chyba na tom, co neviděl?* Schovej pětinu dat do trezoru, natrénuj na zbytku, změř na schované pětině. Když je chyba doma maličká a chyba venku velká, nechytil jsi génia — chytil jsi papouška. A vyjde-li ti obojí podezřele nula, počítáš nejspíš chybu na datech, na kterých ses učil; pak neměříš model, měříš svoje vlastní klam z kapitoly čtyři.