AI Research Labs

121 cards · 9 categories

Cheatsheets

Every formula, metric and rule from the course, grouped by category for revision.

Formulas (36)

Evaluation metrics (10)

Loss functions (6)

Data leakage & good practice (7)

Interpreting models (11)

Algorithms step by step (13)

Comparisons (11)

Core concepts (13)

Common traps (14)



Free course with quizzes, progress tracking and Scoopy, an AI tutor:
airesearchlabs.si/learn/ml

Machine Learning Foundations · Edition November 2026 · © 2026 AI Research Labs. Free for personal study.

Formulas

The equations to know by heart, with what each symbol means.

Population risk

$R(f) = \mathbb{E}[L(Y, f(X))]$: expected loss over the unknown population. Cannot be computed.

1.5 Empirical Risk and Population Risk

Empirical risk

$\hat{R}_n(f) = \frac{1}{n}\sum_{i=1}^{n} L(Y_i, f(X_i))$: average loss on the sample.

1.5 Empirical Risk and Population Risk

Generalisation gap

$R(f) - \hat{R}_n(f)$. Estimated with held-out data.

1.5 Empirical Risk and Population Risk

ERM

$\hat{f} = \arg\min_{f \in \mathcal{F}} \frac{1}{n}\sum_i L(y_i, f(x_i))$. OLS = ERM + squared loss; logistic = ERM + cross-entropy.

2.2 Empirical Risk Minimisation (ERM)

Bias–variance decomposition

$\mathbb{E}[(Y - \hat{f})^2] = \text{bias}^2 + \text{Var}(\hat{f}) + \sigma^2$.

2.5 The Bias-Variance Trade-off

Standardisation

$z = \frac{x - \mu}{\sigma}$ with μ, σ from the **training set only**, reused for validation and test.

2.8 Data Leakage and Good ML Practices

Residual

$r_i = y_i - \hat{y}_i$ (observed − fitted). Positive → model under-predicted.

3.1 Regression Fundamentals

Multiple linear regression

$Y = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p + \varepsilon$; prediction $\hat{y}^* = \hat{\beta}_0 + \sum_j \hat{\beta}_j x_j^*$.

3.2 Simple and Multiple Linear Regression

RSS

$\text{RSS} = \sum_i (y_i - \hat{y}_i)^2 = (y - X\beta)^\top (y - X\beta)$.

3.3 Ordinary Least Squares (OLS)

OLS normal equations

$X^\top X \hat{\beta} = X^\top y \Rightarrow \hat{\beta} = (X^\top X)^{-1} X^\top y$ (if invertible).

3.3 Ordinary Least Squares (OLS)

Design matrix shape

X: $n \times (p+1)$, first column ones. β: $(p+1) \times 1$. $X^\top X$: $(p+1) \times (p+1)$.

3.3 Ordinary Least Squares (OLS)

2×2 inverse

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix}^{-1} = \frac{1}{ad-bc}\begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$$

3.3 Ordinary Least Squares (OLS)

Simple regression by hand

$\hat{\beta}_1 = \frac{S_{xy}}{S_{xx}} = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sum(x_i - \bar{x})^2}$, $\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$.

3.4 Calculating Regression Coefficients by Hand

Standard errors

$\hat{\sigma}^2 = \frac{\text{RSS}}{n-p-1}$, $\text{SE}(\hat{\beta}_j) = \sqrt{\hat{\sigma}^2 [(X^\top X)^{-1}]_{jj}}$. Simple: $\sqrt{\hat{\sigma}^2 / S_{xx}}$.

3.8 Standard Errors, Confidence Intervals and p-values

t-test & CI

$t = \hat{\beta}_j / \mathrm{SE}$, df $= n - p - 1$. CI: $\hat{\beta}_j \pm t_{1-\alpha/2}\,\mathrm{SE}$ (≈ ±2 SE).

3.8 Standard Errors, Confidence Intervals and p-values

VIF

$\mathrm{VIF}_j = \frac{1}{1-R_j^2}$, R_j^2 from X_j on other predictors. Worry > 5–10.

3.9 Multicollinearity and VIF

Odds & log-odds

odds $= \frac{p}{1-p}$; logit $= \ln \frac{p}{1-p}$. $p = \frac{\text{odds}}{1+\text{odds}}$.

4.2 Odds, Log-odds and the Sigmoid Function

Sigmoid

$\sigma(z) = \frac{1}{1+e^{-z}}$. $z = 0 \to 0.5$; $z = \pm 2.2 \to 0.9/0.1$.

4.2 Odds, Log-odds and the Sigmoid Function

Decision boundary

Predict 1 if $\hat{p} > t \iff \hat{\beta}_0 + \hat{\beta}^\top x > \ln \frac{t}{1-t}$ (0 when $t = 0.5$). Linear.

4.3 The Logistic Regression Model

Logistic log-likelihood

$\ell(\beta) = \sum_i [y_i \ln P_i + (1-y_i)\ln(1-P_i)]$. Maximise ⇔ minimise cross-entropy.

4.4 Maximum Likelihood Estimation

Wald test

$z = \hat{\beta}/\mathrm{SE}$, reject if $|z| > 1.96$. OR CI: $e^{\hat{\beta} \pm 1.96\mathrm{SE}}$.

4.4 Maximum Likelihood Estimation

Cost-optimal threshold

Alert if $p > \frac{c_{FP}}{c_{FP}+c_{FN}}$. Costlier misses → lower threshold.

4.9 Threshold Selection and Classification Costs

Expected cost

$\mathrm{Cost}(t) = c_{FP}\,FP(t) + c_{FN}\,FN(t)$; choose t on validation data.

4.9 Threshold Selection and Classification Costs

Regularised objective

$\min_\beta \{\text{training error}(\beta) + \lambda\,\text{Penalty}(\beta)\}$, $\lambda \geq 0$.

5.1 Why Regularisation is Needed

Model counts

Best subset: 2^p. Stepwise: $1 + \frac{p(p+1)}{2}$.

5.3 Best Subset and Stepwise Selection

Ridge

$\min\ \mathrm{RSS} + \lambda \sum \beta_j^2$; $\hat{\beta} = (X^\top X + \lambda I)^{-1} X^\top y$. Orthonormal: $\hat{\beta}^{\mathrm{OLS}}/(1+\lambda)$.

5.4 Ridge Regression (L2)

LASSO

$\min\ \mathrm{RSS} + \lambda \sum |\beta_j|$. Orthonormal: $\mathrm{sign}(b)\max(|b| - \lambda/2, 0)$ (soft threshold).

5.5 LASSO Regression (L1)

Norms

$\|\beta\|_0$ = #non-zeros; $\|\beta\|_1 = \sum |\beta_j|$; $\|\beta\|_2^2 = \sum \beta_j^2$ (intercept excluded).

5.8 Mathematical Objectives and Worked Calculations

Regularised logistic

$\min\{-\ell(\beta) + \lambda \sum \beta_j^2\}$ (ridge) or $+\lambda \sum |\beta_j|$ (LASSO).

5.8 Mathematical Objectives and Worked Calculations

K-means objective

$J = \sum_k \sum_{i \in C_k} \|x_i - \mu_k\|^2$ (WCSS). Local minimum only.

6.1 K-means Clustering

Distances

Euclidean $\sqrt{\sum(a_j - b_j)^2}$; Manhattan $\sum |a_j - b_j|$. K-means compares squared Euclidean.

6.2 Distance Calculations and Cluster Assignment

SVM decision rule

$f(x) = w^\top x + b$; predict $\text{sign}(f(x))$. Labels $y \in \{-1, +1\}$.

6.4 SVM Fundamentals

SVM geometry

Boundary $w^\top x + b = 0$; margins $= \pm 1$. Distance $\frac{|w^\top x + b|}{\|w\|}$. Width $\frac{2}{\|w\|}$.

6.5 Margins and Support Vectors

Hard-margin SVM

$\min \frac{1}{2}\|w\|^2$ s.t. $y_i(w^\top x_i + b) \geq 1$.

6.6 Hard-margin and Soft-margin SVM

Soft-margin SVM

$\min \frac{1}{2}\|w\|^2 + C\sum_i \xi_i$, $\xi_i = \max(0, 1 - y_i f(x_i))$. ≡ hinge + L2, $\lambda \propto 1/C$.

6.6 Hard-margin and Soft-margin SVM

Kernels

Linear $x^\top x'$ · Polynomial $(x^\top x' + c)^d$ · RBF $e^{-\gamma\|x - x'\|^2}$.

6.7 The C Parameter and Kernels

Evaluation metrics

How to measure a model, and when each measure misleads.

MAE / RMSE

MAE $= \frac{1}{n}\sum|e_i|$ (robust). RMSE $= \sqrt{\frac{1}{n}\sum e_i^2}$ (punishes big errors). Both in units of Y.

3.7 R^2, Adjusted R^2, RMSE and Other Metrics

R^2

$R^2 = 1 - \frac{\text{RSS}}{\text{TSS}}$, $\text{TSS} = \sum(y_i - \bar{y})^2$. Share of the mean-baseline's error removed.

3.7 R^2, Adjusted R^2, RMSE and Other Metrics

Adjusted R^2

$1 - \frac{\text{RSS}/(n-p-1)}{\text{TSS}/(n-1)}$. Falls when a useless predictor is added.

3.7 R^2, Adjusted R^2, RMSE and Other Metrics

Out-of-sample R^2

Baseline is $\bar{y}_{\text{train}}$ on test data. Can be negative.

3.7 R^2, Adjusted R^2, RMSE and Other Metrics

Confusion matrix

Rows actual, columns predicted. TP hit · FN **miss** · FP **false alarm** · TN. FP = Type I, FN = Type II.

4.6 The Confusion Matrix

Sensitivity / recall / TPR

$\frac{TP}{TP+FN}$: share of actual positives caught.

4.7 Accuracy, Precision, Recall and Specificity

Specificity / TNR

$\frac{TN}{TN+FP}$: share of actual negatives cleared. FPR $= 1-$ spec.

4.7 Accuracy, Precision, Recall and Specificity

Precision / PPV

$\frac{TP}{TP+FP}$: share of flagged cases truly positive. Depends on prevalence.

4.7 Accuracy, Precision, Recall and Specificity

Accuracy & F1

Acc $= \frac{TP+TN}{n}$. $F_1 = \frac{2PR}{P+R}$.

4.7 Accuracy, Precision, Recall and Specificity

ROC & AUC

ROC: TPR vs FPR over all thresholds. AUC = P(random positive scored above random negative). 0.5 random, 1 perfect.

4.10 ROC Curve and AUC

Loss functions

What each loss punishes and where it is used.

Squared loss (L2)

$(y - \hat{y})^2$. Punishes big errors; outlier-sensitive; OLS, ridge, LASSO.

2.1 Loss Functions and Risk

Absolute loss (L1)

$|y - \hat{y}|$. Linear; robust to outliers; MAE.

2.1 Loss Functions and Risk

Cross-entropy / log loss

$-[y \ln p + (1 - y) \ln(1 - p)]$. Logistic regression. Confidently wrong = huge loss.

2.1 Loss Functions and Risk

0–1 loss

$1[y \neq \hat{y}]$. Averages to the error rate; not smooth.

2.1 Loss Functions and Risk

Log-loss values

$-\ln p$: 0.99→0.01, 0.9→0.105, 0.5→0.693, 0.1→2.303, 0.01→4.6.

4.5 Cross-entropy / Log-loss

Hinge loss

$\max(0, 1 - yf(x))$. 0 beyond the margin; 0–1 inside; >1 if misclassified.

6.6 Hard-margin and Soft-margin SVM

Data leakage & good practice

How evaluation goes wrong, and the rule that fixes each case.

Train / validation / test

Train: fit parameters. **Validation**: choose models & hyperparameters. **Test**: final estimate, used **once**.

2.6 Training, Validation and Test Sets

CV with preprocessing

Scaling, imputation, feature selection and oversampling happen **inside each fold**, on that fold's training part only.

2.7 Cross-validation

The six leaks

1 Test reuse · 2 Preprocess before split · 3 Feature selection before CV · 4 Related records across splits · 5 Time leakage · 6 Target leakage.

2.8 Data Leakage and Good ML Practices

Six golden rules

Split before preprocessing. Never tune on test. Select features inside folds. Keep related rows together. Respect time order. Use only prediction-time features.

2.8 Data Leakage and Good ML Practices

One test for every leak

Did anything from held-out data, the future, or the outcome influence a calculation or decision before the final score? If yes → leak.

2.8 Data Leakage and Good ML Practices

Oversampling rule

Split first. Oversample only the training part (inside each CV fold). Evaluate on the original class ratio.

4.11 Class Imbalance

Regularisation & leakage

Standardise within training folds. Never choose λ on the test set.

5.9 Selecting λ Using Cross-validation

Interpreting models

Ready-made sentences for explaining coefficients, p-values and plots.

Diagnose from errors

Train low, val high → **overfit**. Both high → **underfit**. Train & val good, test bad → shift / leakage / evaluation issue.

2.3 Underfitting and Overfitting

Coefficient sentence

"Holding the other predictors fixed, a one-unit increase in X_j is associated with a $\hat{\beta}_j$ change in predicted Y."

3.5 Interpreting Coefficients

Residual plot patterns

Random cloud → OK. Curve → missing nonlinearity. Fan → heteroskedasticity. Waves in time → autocorrelation. Heavy tails → outliers / non-normal.

3.6 Residuals and Regression Assumptions

p-value sentence

"If the true coefficient were zero (and assumptions hold), a t-statistic at least this extreme would occur with probability p."

3.8 Standard Errors, Confidence Intervals and p-values

Multicollinearity effects

Inflated SEs, unstable/wrong-sign coefficients. Predictions often fine. Fix: drop/combine, more data, ridge.

3.9 Multicollinearity and VIF

Logistic coefficient

β_j: change in log-odds per unit X_j. e^{β_j}: odds ratio (multiplies the odds). Not a constant change in probability.

4.3 The Logistic Regression Model

Threshold effects

Lower t → more flagged → sens ↑, spec ↓, FN ↓, FP ↑, precision usually ↓. Raise t → the reverse.

4.8 Sensitivity and Specificity Trade-offs

Ridge properties

Shrinks, never zero. Always solvable. Shares credit among correlated predictors. Bias ↑, variance ↓ with λ.

5.4 Ridge Regression (L2)

λ regimes

λ=0: OLS, min training error. Small: slight shrink, test often ↓. Middle: best (CV). Large: high bias. ∞: predicts $\bar{y}$.

5.6 Understanding λ (Lambda)

K-means failure modes

Unscaled features · bad starts (local optima) · wrong K · non-round clusters · outliers · categorical data.

6.3 Updating Centroids and Convergence

Effect of C

Large C: narrow margin, few violations, low bias/high variance. Small C: wide margin, high bias/low variance. C ~ 1/λ.

6.7 The C Parameter and Kernels

Algorithms step by step

The procedures you may be asked to run by hand.

ML workflow

Define problem → explore data → **split** → fit candidates on train → compare on validation/CV → inspect & revise → test **once** → deploy & monitor.

1.3 The ML Modelling Process

K-fold CV

1) Split into K folds. 2) For each fold: train on $K-1$, validate on it. 3) Average the K errors. 4) Pick the best candidate; re-fit on all training data.

2.7 Cross-validation

"Is this approach reasonable?"

1) Identify the problem. 2) Name the leak. 3) Explain the consequence (optimistic estimate). 4) Recommend the fix.

2.8 Data Leakage and Good ML Practices

Slope by hand

1) Means $\bar{x}, \bar{y}$. 2) Deviations. 3) Products & squares. 4) Sum → S_{xy}, S_{xx}. 5) Divide; then intercept. 6) Check residuals sum to 0.

3.4 Calculating Regression Coefficients by Hand

AUC by counting

For every (positive, negative) pair, score 1 if the positive ranks higher, 0.5 if tied. AUC = total / (#pos × #neg).

4.10 ROC Curve and AUC

Imbalance toolkit

Right metrics + majority baseline · cost-based threshold · oversample/undersample training data · class weights.

4.11 Class Imbalance

Forward stepwise

Start M_0. Repeatedly add the predictor that most lowers RSS → $M_1..M_p$. Choose the size by validation/CV.

5.3 Best Subset and Stepwise Selection

Best subset

For each k, best of $\binom{p}{k}$ models by RSS → M_k. Pick among M_k by CV / adj. R^2 / AIC / BIC. Cost 2^p.

5.3 Best Subset and Stepwise Selection

Penalised objective by hand

1) Compute RSS (or $-\ell$). 2) Compute the norm of the slopes. 3) Add λ × norm. Compare candidates on the same λ.

5.8 Mathematical Objectives and Worked Calculations

Choosing λ

Split off test → log-spaced λ grid → K-fold CV (standardise inside folds) → pick min CV or 1-SE → refit on all training → test once.

5.9 Selecting λ Using Cross-validation

K-means

1) Initialise K centroids. 2) Assign each point to the nearest centroid. 3) Move each centroid to its cluster mean. 4) Repeat until assignments stop changing.

6.1 K-means Clustering

Choosing K

Not by min J (falls with K). Use the elbow, silhouette, stability and usefulness.

6.3 Updating Centroids and Convergence

SVM by hand (2 support vectors)

Use symmetry for the direction of w; set $f = -1$ at the negative SV and $f = +1$ at the positive SV; solve; width $= 2/\|w\|$.

`6.5 Margins and Support Vectors`

Comparisons

Side-by-side differences between methods that are easy to mix up.

Supervised vs unsupervised

Supervised: have X and Y; learn $X \to Y$ (regression, classification). **Unsupervised:** only X; find structure (clustering).

1.2 Types of Machine Learning

Parametric vs non-parametric

Parametric (linear/logistic regression): fixed shape, few parameters, less data, interpretable, risk of bias. **Non-parametric** (K-NN): flexible, needs more data, risk of variance.

1.2 Types of Machine Learning

Loss vs metric

Loss: minimised during fitting (must be convenient). **Metric**: judges the final model for the task (AUC, cost, RMSE).

2.1 Loss Functions and Risk

Simple vs flexible

Simple: high bias, low variance, underfits. Flexible: low bias, high variance, overfits.

2.5 The Bias–Variance Trade-off

Linear vs logistic regression

Linear: number, models $\mathbb{E}[Y|X]$, OLS, RMSE/R^2. Logistic: class via $P(Y=1|X) \in (0,1)$, linear in log-odds, MLE, confusion matrix/AUC.

4.1 Regression versus Classification

OLS vs MLE fitting

OLS: minimise RSS, closed form. Logistic MLE: maximise log-likelihood, numerical (concave, no bad local optima).

4.4 Maximum Likelihood Estimation

Ridge vs LASSO in one line

Ridge divides (never zero); LASSO subtracts (exact zeros → selection).

5.5 LASSO Regression (L1)

Subset vs ridge vs LASSO

Zeros: yes / no / yes. Closed form: no / yes / no. Cost: 2^p / easy / moderate. Correlated: arbitrary / shares / picks one.

5.7 Ridge versus LASSO

K-means vs K-NN

K-means: unsupervised, K = clusters. K-NN: supervised, K = neighbours.

6.1 K-means Clustering

Logistic regression vs linear SVM

Labels 0/1 vs ±1 · cross-entropy vs hinge · probability vs score · all points vs support vectors · λ vs C · features vs kernels.

6.4 SVM Fundamentals

Complexity knobs, all models

More complex: ↑ degree, ↑ #features, ↓ λ, ↓ K (K-NN), ↑ C, ↑ γ (RBF), ↑ tree depth.

6.7 The C Parameter and Kernels

Core concepts

One-line definitions of the big ideas.

Machine learning

Learning a prediction rule f from data so that $\hat{y} = f(X)$ predicts Y well on **new** data.

1.1 What is Machine Learning?

Features vs target

Features / predictors X: the inputs. **Target / response / label** Y: what we predict.

1.1 What is Machine Learning?

Regression vs classification

Regression predicts a **number**. Classification predicts a **category**.

1.2 Types of Machine Learning

Hyperparameter

A setting chosen by you, not fitted from data (degree, depth, K, λ, C). Choose it on validation data or by CV.

1.3 The ML Modelling Process

No Free Lunch

Averaged over all problems no algorithm wins. In practice: no universally best model, so compare candidates on relevant data and state your assumptions.

1.4 Fundamental Limits of Machine Learning

Generalisation gap drivers

↑ complexity → ↑ gap. ↑ data → ↓ gap. ↑ regularisation → ↓ gap. ↑ noise → ↑ gap.

2.4 Generalisation and the Generalisation Gap

Typical splits

60:20:20 or 50:25:25 (train:val:test); train:val 50–80% / 20–50%. A design choice.

2.6 Training, Validation and Test Sets

Signal + noise

$Y = f(X) + \varepsilon$, $\mathbb{E}[\varepsilon] = 0$. Linear regression: $f(X) = \beta_0 + \sum_j \beta_j X_j$.

3.1 Regression Fundamentals

OLS noise assumptions

Mean zero · constant variance · independent · (roughly normal for t-tests).

3.6 Residuals and Regression Assumptions

When to regularise

$p \approx n$ or $p > n$ · correlated predictors · noisy data.

5.1 Why Regularisation is Needed

Complexity dials

#features, |β|, degree, tree depth, small K (K-NN), large K (K-means), large C (SVM).

5.2 Model Complexity and Feature Selection

Why LASSO gives zeros

L1 constraint = diamond with corners on the axes; RSS ellipses usually touch a corner first.

5.7 Ridge versus LASSO

Support vectors

Points on (or inside) the margin. They alone determine the boundary.

6.5 Margins and Support Vectors

Common traps

Mistakes that cost marks, and the correct statement.

NFL misreading

Wrong: "all algorithms do equally well on my data". Right: it is an average over every possible problem; real problems have structure.

1.4 Fundamental Limits of Machine Learning

Training error picks complexity

Wrong: choose the model with the lowest training error. Right: choose on validation data or by CV.

2.3 Underfitting and Overfitting

"Variance" means...

Variance of the **fitted model** across training sets, not the variance of the data.

2.5 The Bias–Variance Trade-off

"Linear" regression

Linear in the **coefficients**, not the inputs. X^2, $\log X$, $X_1 X_2$ are allowed.

3.2 Simple and Multiple Linear Regression

Comparing coefficients

Raw sizes depend on units. Standardise predictors first. And association ≠ causation.

3.5 Interpreting Coefficients

R^2 misreadings

Not "% correct". High in-sample ≠ good prediction. Low ≠ useless predictors.

3.7 R^2, Adjusted R^2, RMSE and Other Metrics

p-value misreadings

Not P(H_0 true). Not effect size. Large p ≠ no effect.

3.8 Standard Errors, Confidence Intervals and p-values

Odds ratio 2 ≠ "twice as likely"

It doubles the **odds**. The probability ratio is smaller unless p is small.

4.3 The Logistic Regression Model

Sign of the log-likelihood

ℓ is maximised (≤ 0). Cross-entropy $= -\ell/n$ is minimised (≥ 0).

4.5 Cross-entropy / Log-loss

High accuracy, rare class

Compare with the majority-class baseline. Report sensitivity, specificity, precision.

4.7 Accuracy, Precision, Recall and Specificity

AUC misreadings

AUC 0.9 ≠ 90% accuracy. It doesn't choose a threshold or account for costs.

4.10 ROC Curve and AUC

Select by p-values

Dropping $p > 0.05$ and refitting ignores selection uncertainty and over-states significance. Use CV stepwise or LASSO.

5.2 Model Complexity and Feature Selection

Lowest training error?

Always at λ = 0. Regularisation never beats OLS on training RSS.

5.6 Understanding λ (Lambda)

Unscaled features

The largest-range feature dominates distances. Standardise before K-means, K-NN, SVM.

6.2 Distance Calculations and Cluster Assignment