# Optimality Conditions in Several Variables

Last modified: 2026-11-24 09:19:45 (PST)

## Optimality conditions in several variables

> **Source**
>
> This section is adapted from the multivariate calculus and optimality-condition parts of Zhou [1], used under the MIT License (see the license text in Linear Algebra). The source states the second-order Taylor approximation and the optimality conditions without proof; here they are proved from the one-variable Taylor theorem, and the approximation is made exact by evaluating the Hessian at a point between $\underset{\sim}{z}$ and $\underset{\sim}{z} + \underset{\sim}{h}$. The source calls points with zero gradient critical points; here they are stationary points, because Definition 2 in Critical Points and Optimization gives "critical point" a wider meaning. The source states these results for differentiable functions without saying how smooth; here the first-order results assume continuous first partial derivatives, and the second-order results continuous second partial derivatives. These parts of the source are not part of this section:
>
> - its remark that maximizing $f$ is minimizing $-f$, and its classification of constraints
> - its figure of local, strict local and global minima
> - its one-variable Taylor example and its plots
> - matrix calculus
> - convexity
> - Lagrange multipliers
> - Newton's method and gradient descent

A local minimizer of a differentiable function on an open interval is a flat point (Definition 1 in Critical Points and Optimization), and, at a flat point, a positive second derivative guarantees a strict local minimum. This section extends both facts to a function $f$ of a $p \times 1$ vector $\underset{\sim}{x}$: the gradient $\frac{\partial}{\partial \underset{\sim}{x}} f(\underset{\sim}{x})$ takes the place of the first derivative, and a $p \times p$ matrix of second partial derivatives takes the place of the second; both facts are proved in this section, for every $p \geq 1$. Throughout, $\frac{\partial}{\partial \underset{\sim}{x}} f(\underset{\sim}{z})$ means the gradient evaluated at $\underset{\sim}{x} = \underset{\sim}{z}$.

**Theorem 0.1** (A minimizer has derivative zero)**.** *Let $g$ be a real-valued function on an open interval containing $c$, and let $g$ be differentiable at $c$. If $c$ is a local minimizer of $g$, then $g'(c) = 0$. [2]*

Proof

*Proof.* Take $\delta > 0$ as in the definition of a local minimizer, shrunk if needed so that $(c-\delta, c+\delta)$ lies inside the interval; then $g(c) \leq g(c+h)$ whenever $|h| < \delta$. For $0 < h < \delta$, the difference quotient is

$$\frac{g(c+h) - g(c)}{h} \geq 0,$$

because its numerator is at least 0 and its denominator is positive. As $h \to 0$ from the right, these quotients tend to $g'(c)$ (the two-sided limit exists, so the one-sided limit equals it), and a limit of numbers that are all at least 0 is at least 0; so $g'(c) \geq 0$. For $-\delta < h < 0$, the numerator is still at least 0 but the denominator is negative, so the quotient is at most 0, and in the same way, with $h \to 0$ from the left, $g'(c) \leq 0$. Together, $g'(c) = 0$. □

**Example 0.1** (Applying the test, and its limits)**.**

- $g(x) = (x-2)^2$ has its minimizer at $c = 2$, and

$$\begin{aligned} g'(2) &= 2(2-2) \\ &= 0. \end{aligned}$$

- The converse fails: $g(x) = x^3$ has

$$\begin{aligned} g'(0) &= 3 \cdot 0^2 \\ &= 0, \end{aligned}$$

  but 0 is not a local minimizer, since

$$\begin{aligned} g(-h) &= -h^3 \\ &< 0 \\ &= g(0) \end{aligned}$$

  for every $h > 0$.

- Differentiability is needed: $g(x) = |x|$ has its minimizer at 0, but the difference quotient $|h|/h$ is 1 for $h > 0$ and $-1$ for $h < 0$, so it has no limit and $g'(0)$ does not exist.

**Definition 0.1** (Stationary point)**.** Let $f$ be a scalar-valued function of a $p \times 1$ vector $\underset{\sim}{x}$. A point $\underset{\sim}{z}$ at which the gradient exists and

$$\frac{\partial}{\partial \underset{\sim}{x}} f(\underset{\sim}{z}) = \underset{\sim}{0}_{p\times 1}$$

is a **stationary point** of $f$.

**Example 0.2** (A stationary point, and a point that is not one)**.** Let $f(\underset{\sim}{x}) = x_1^2 + x_2^2 - 2x_1$. Its gradient is

$$\nabla f(\underset{\sim}{x}) = (2x_1 - 2,\ 2x_2)^\top,$$

which is $\underset{\sim}{0}_{2\times 1}$ exactly when $x_1 = 1$ and $x_2 = 0$. So $(1, 0)^\top$ is its only stationary point, and $\underset{\sim}{0}$ is not one: $\nabla f(\underset{\sim}{0}_{2\times 1}) = (-2, 0)^\top$.

Remark (Stationary points in one variable)

*Remark* 0.1 (Stationary points in one variable). For $p = 1$, a stationary point is a flat point (Definition 1 in Critical Points and Optimization).

**Theorem 0.2** (First-order necessary condition)**.** *Let $f : \mathbb{R}^p \to \mathbb{R}$. If $\underset{\sim}{x}^*$ is a local minimizer of $f$ and the gradient of $f$ exists at $\underset{\sim}{x}^*$, then $\underset{\sim}{x}^*$ is a stationary point of $f$ (Definition 0.1).*

Proof

*Proof.* Take $\delta > 0$ as in the definition of a local minimizer. Let $i \in \{1, \dots, p\}$, let $\underset{\sim}{e}_i$ be the vector with 1 in entry $i$ and 0 elsewhere, and let $g_i(t) \overset{\text{def}}{=} f(\underset{\sim}{x}^* + t\,\underset{\sim}{e}_i)$ for $t \in \mathbb{R}$. For $|t| < \delta$,

$$\begin{aligned} \|(\underset{\sim}{x}^* + t\,\underset{\sim}{e}_i) - \underset{\sim}{x}^*\| &= |t|\,\|\underset{\sim}{e}_i\| \\ &= |t| \\ &< \delta, \end{aligned}$$

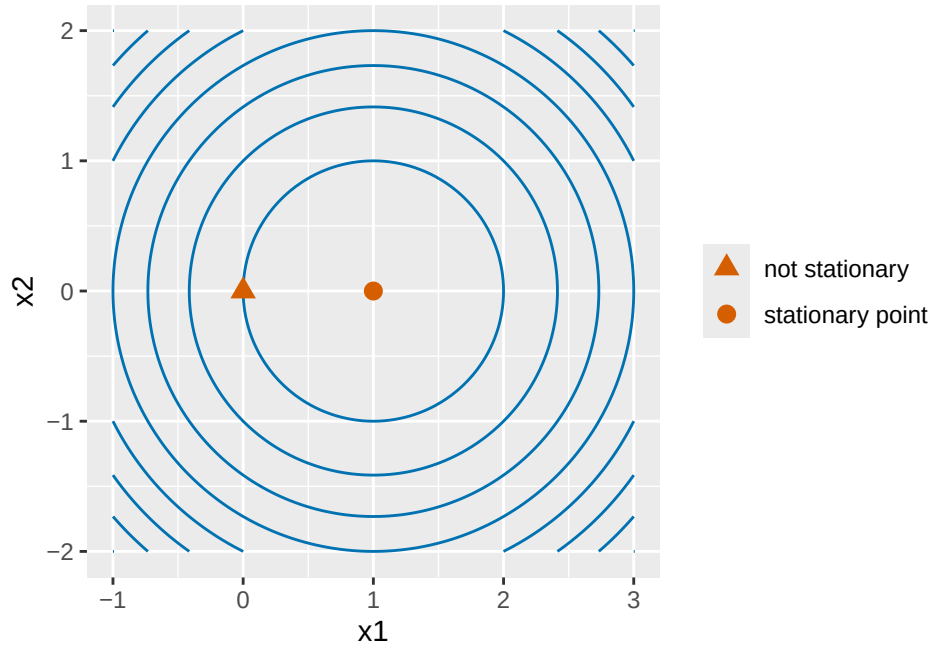
so

$$\begin{aligned} g_i(0) &= f(\underset{\sim}{x}^*) \\ &\le f(\underset{\sim}{x}^* + t\,\underset{\sim}{e}_i) \\ &= g_i(t)\,: \end{aligned}$$

0 is a local minimizer of $g_i$. Moving $\underset{\sim}{x}$ from $\underset{\sim}{x}^*$ along $\underset{\sim}{e}_i$ changes only $x_i$, so $g_i'(0)$ is the

**R**

```r
sp_x1_fs <- seq(-1, 3, length.out = 101)
sp_x2_fs <- seq(-2, 2, length.out = 101)
sp_grid_fs <- expand.grid(x1 = sp_x1_fs, x2 = sp_x2_fs)
sp_grid_fs$value <- sp_grid_fs$x1^2 + sp_grid_fs$x2^2 - 2 * sp_grid_fs$x1
sp_marks_x1_fs <- c(1, 0)
sp_marks_kind_fs <- c("stationary point", "not stationary")
sp_marks_fs <- tibble::tibble(
  x1 = sp_marks_x1_fs, x2 = 0, kind = sp_marks_kind_fs
)
sp_aes_fs <- ggplot2::aes(x1, x2, z = value)
sp_marks_aes_fs <- ggplot2::aes(x1, x2, shape = kind, z = NULL)
sp_shapes_fs <- c("stationary point" = 16, "not stationary" = 17)
ggplot2::ggplot(sp_grid_fs, sp_aes_fs) +
  ggplot2::geom_contour(colour = "#0072B2") +
  ggplot2::geom_point(
    data = sp_marks_fs, mapping = sp_marks_aes_fs,
    colour = "#D55E00", size = 3, inherit.aes = FALSE
  ) +
  ggplot2::scale_shape_manual(values = sp_shapes_fs) +
  ggplot2::coord_fixed() +
  ggplot2::labs(x = "x1", y = "x2", shape = NULL)
```






**Python**

```python
import matplotlib.pyplot as plt
import numpy as np

sp_x1_fs = np.linspace(-1, 3, 101)
sp_x2_fs = np.linspace(-2, 2, 101)
```

**R**

*(This 3D surface is interactive; see the HTML or slide version of this page to view it.)*

**Python**

```
import matplotlib.pyplot as plt
import numpy as np

sp_x1_3d = np.linspace(-1, 3, 101)
sp_x2_3d = np.linspace(-2, 2, 101)
sp_x1_grid_3d, sp_x2_grid_3d = np.meshgrid(sp_x1_3d, sp_x2_3d)
sp_value_3d = sp_x1_grid_3d**2 + sp_x2_grid_3d**2 - 2 * sp_x1_grid_3d
fig = plt.figure(figsize=(6, 4.5))
ax = fig.add_subplot(projection="3d")
ax.plot_surface(sp_x1_grid_3d, sp_x2_grid_3d, sp_value_3d,
                color="#0072B2", alpha=0.6)
ax.scatter([1], [0], [-1], marker="o", color="#D55E00", s=50,
           label="stationary point")
ax.scatter([0], [0], [0], marker="D", color="#D55E00", s=50,
           label="not stationary")
ax.set_xlabel("x1")
ax.set_ylabel("x2")
ax.set_zlabel("f(x)")
ax.legend(loc="upper right")
plt.show();
```

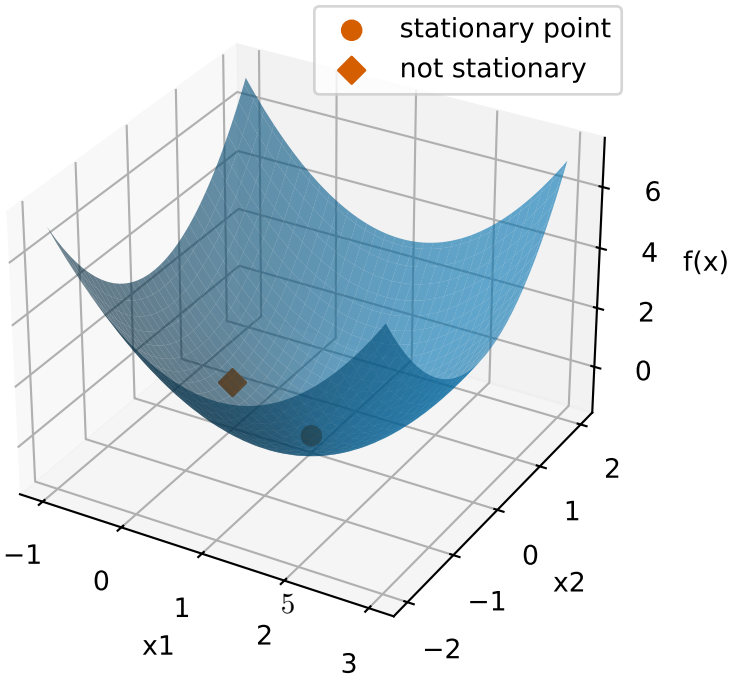



Figure 2: The graph of the function of Example 0.2, the surface whose level curves are in Figure 1. The surface is flat, with gradient 0, only at the bottom of the bowl

partial derivative $\frac{\partial}{\partial x_i} f(\underset{\sim}{x}^*)$, which exists because the gradient does. By Theorem 0.1,

$$\begin{aligned} \frac{\partial}{\partial x_i} f(\underset{\sim}{x}^*) &= g_i'(0) \\ &= 0. \end{aligned}$$

The equation $g_i'(0) = 0$ holds for every $i$, so every entry of the gradient at $\underset{\sim}{x}^*$ is 0. □

**Example 0.3** (Using the condition to locate, and to rule out, minimizers)**.**

- For $f(\underset{\sim}{x}) = x_1^2 + x_2^2 - 2x_1$ of Example 0.2, any local minimizer must be the stationary point $(1, 0)^\top$. It is one: completing the square,

$$\begin{aligned} f(\underset{\sim}{x}) &= (x_1 - 1)^2 + x_2^2 - 1 \\ &\geq -1 \\ &= f(1, 0). \end{aligned}$$

- For $f(\underset{\sim}{x}) = e^{2x_1 + x_2} - x_1$ of the Hessian example, the second entry of $\nabla f(\underset{\sim}{x})$ is $e^{2x_1 + x_2} > 0$, so $f$ has no stationary point, and so no local minimizer.
- The converse fails: $f(\underset{\sim}{x}) = x_1^2 - x_2^2$ has gradient $(2x_1,\ -2x_2)^\top$, so $\underset{\sim}{0}$ is a stationary point, but

$$\begin{aligned} f(0, t) &= -t^2 \\ &< 0 \\ &= f(\underset{\sim}{0}) \end{aligned}$$

  for every $t \neq 0$, so $\underset{\sim}{0}$ is not a local minimizer.

**Example 0.4** (A stationary point of the sum of squared errors, on real data)**.** The Auto data record the horsepower and fuel economy (miles per gallon) of 392 car models sold between 1970 and 1982. They come from the StatLib library at Carnegie Mellon University and ship with the book *An Introduction to Statistical Learning*, in the Python package `ISLP` (`ISLP.load_data("Auto")`) and the R package `ISLR2` (`ISLR2::Auto`). This site has a copy at `data/auto.csv`, and both code tabs below read that file.
Let $y_i$ be the miles per gallon and $x_i$ the horsepower of car $i$. To predict miles per gallon from horsepower with a line, write $\beta = (\beta_0, \beta_1)^\top$ and let $\mathbf{X}$ be the matrix whose row $i$ is $(1, x_i)$. The sum of squared errors is

$$f(\underset{\sim}{\beta}) = \|\underset{\sim}{y} - \mathbf{X}\underset{\sim}{\beta}\|^2,$$

a scalar-valued function of the $2 \times 1$ vector $\underset{\sim}{\beta}$. Its gradient is

$$\nabla f(\underset{\sim}{\beta}) = -2\,\mathbf{X}^\top(\underset{\sim}{y} - \mathbf{X}\underset{\sim}{\beta}).$$

By Theorem 0.2, a minimizer of $f$ must make this gradient $\underset{\sim}{0}$, which is the equation $\mathbf{X}^\top\mathbf{X}\beta = \mathbf{X}^\top y$. The code below solves that equation and evaluates the gradient at the solution and at $\underset{\sim}{\beta} = \underset{\sim}{0}$.

**Python**

```
import numpy as np
import pandas as pd

auto = pd.read_csv("data/auto.csv")
y = auto["mpg"].to_numpy(dtype=float)
X = np.column_stack([np.ones(len(auto)), auto["horsepower"].to_numpy(dtype=float)])

def grad(beta):
    return -2 * X.T @ (y - X @ beta)

beta_hat = np.linalg.solve(X.T @ X, X.T @ y)
grad_hat = grad(beta_hat)
grad_zero = grad(np.zeros(2))

pd.DataFrame(
    {"beta": [beta_hat, np.zeros(2)], "gradient": [grad_hat, grad_zero]},
    index=["solution", "zero vector"],
)
#>                                beta                gradient
#> solution     [39.935861, -0.157845]              [0.0, 0.0]
#> zero vector              [0.0, 0.0]  [-18381.6, -1737437.6]
```

A library fit gives the same coefficients:

```
import numpy as np

# np.polyfit returns the slope first, so reverse to (intercept, slope).
beta_polyfit = np.polyfit(auto["horsepower"], auto["mpg"], deg=1)[::-1]
beta_polyfit
#> array([39.93586102, -0.15784473])
```

**R**

```
auto <- read.csv("data/auto.csv")
y <- auto$mpg
X <- cbind(1, auto$horsepower)

grad <- function(beta) -2 * t(X) %*% (y - X %*% beta)

beta_hat <- solve(t(X) %*% X, t(X) %*% y)
grad_hat <- grad(beta_hat)
grad_zero <- grad(c(0, 0))

list(
  solution = list(beta = c(beta_hat), gradient = c(grad_hat)),
  zero_vector = list(beta = c(0, 0), gradient = c(grad_zero))
)
#> $solution
#> $solution$beta
#> [1] 39.9358610 -0.1578447
#>
#> $solution$gradient
#> [1] 9.471535e-12 6.662049e-11
#>
#>
#> $zero_vector
#> $zero_vector$beta
#> [1] 0 0
#>
#> $zero_vector$gradient
#> [1]   -18381.6 -1737437.6
```

A library fit gives the same coefficients:

```
coef(lm(mpg ~ horsepower, data = auto))
#> (Intercept)  horsepower
#>  39.9358610  -0.1578447
```

The solution is an intercept of 39.9 miles per gallon and a slope of -0.158 miles per gallon per unit of horsepower. The largest entry of the gradient there is 6.7e-11, zero up to rounding error, against 1,737,438 at the zero vector. So the solution is a stationary point (Definition 0.1), and the zero vector is not. The library fits above agree with the solution of that equation.

**Theorem 0.3** (Taylor's theorem with a second-order remainder)**.** *Let $a < b$, and let $g$ be a real-valued function on $[a, b]$ whose derivative $g'$ is continuous on $[a, b]$ and whose second derivative $g''$ exists at every point of $(a, b)$. Then there is a $\tau \in (a, b)$ with [3]*

$$g(b) = g(a) + g'(a)\,(b - a) + \frac{1}{2}\,g''(\tau)\,(b - a)^2.$$

This is the case $n = 2$ of [4], Theorem 5.15, whose proof uses the mean value theorem, which these notes do not develop.

**Example 0.5** (The remainder point for $e^x$)**.** Take $g(x) = e^x$ on $[0, 1]$, so

$$\begin{aligned} g' &= g'' \\ &= g. \end{aligned}$$

Theorem 0.3 says $e = 1 + 1 + \frac{1}{2}\,e^\tau$ for some $\tau \in (0, 1)$. Solving,

$$\begin{aligned} e^\tau &= 2(e - 2) \\ &\approx 2 \times 0.71828 \\ &= 1.43656, \end{aligned}$$

so $\tau = \log 1.43656 \approx 0.362$, where log is the natural logarithm. This $\tau$ is in $(0, 1)$.

**Theorem 0.4** (Second-order Taylor theorem in several variables)**.** *Let $f : \mathbb{R}^p \to \mathbb{R}$ have first and second partial derivatives that are continuous on $\mathbb{R}^p$, and let $\underset{\sim}{z}, \underset{\sim}{h} \in \mathbb{R}^p$. Then there is a $\tau \in (0, 1)$ with [3]*

$$f(\underset{\sim}{z} + \underset{\sim}{h}) = f(\underset{\sim}{z}) + \left(\frac{\partial}{\partial \underset{\sim}{x}} f(\underset{\sim}{z})\right)^\top \underset{\sim}{h} + \frac{1}{2}\,\underset{\sim}{h}^\top \mathbf{H}_f(\underset{\sim}{z} + \tau \underset{\sim}{h})\,\underset{\sim}{h}.$$

Proof

*Proof.* Let $g(t) \overset{\text{def}}{=} f(\underset{\sim}{z} + t\underset{\sim}{h})$ for $t \in \mathbb{R}$, the values of $f$ along the line through $\underset{\sim}{z}$ in the direction $\underset{\sim}{h}$. The inner function $\underset{\sim}{y}(t) = \underset{\sim}{z} + t\underset{\sim}{h}$ has entries $y_j = z_j + th_j$, so $\frac{\partial}{\partial t} y_j = h_j$, and its derivative is the $1 \times p$ matrix $\underset{\sim}{h}^\top$. By the vector chain rule applied with the argument $t$, the inner function $\underset{\sim}{y}(t) = \underset{\sim}{z} + t\underset{\sim}{h}$, and the outer function $f$,

$$\begin{aligned} g'(t) &= \underset{\sim}{h}^\top \frac{\partial}{\partial \underset{\sim}{x}} f(\underset{\sim}{z} + t\underset{\sim}{h}) && \text{(vector chain rule)} \\ &= \sum_{j=1}^{p} h_j \frac{\partial}{\partial x_j} f(\underset{\sim}{z} + t\underset{\sim}{h}). && \text{(matrix product)} \end{aligned}$$

The second step writes out the matrix product. $g'$ is continuous: $t \mapsto \underset{\sim}{z}+t\underset{\sim}{h}$ is continuous, since $\|(\underset{\sim}{z}+s\underset{\sim}{h})-(\underset{\sim}{z}+t\underset{\sim}{h})\| = |s-t|\,\|\underset{\sim}{h}\|$, each $\frac{\partial}{\partial x_j}f$ is continuous by assumption, and compositions, sums and constant multiples of continuous functions are continuous. Each $\frac{\partial}{\partial x_j}f$ has continuous partial derivatives, the second partial derivatives of $f$, so the same chain-rule computation applies to it: $\frac{\partial}{\partial t}\left[\frac{\partial}{\partial x_j}f(\underset{\sim}{z}+t\underset{\sim}{h})\right] = \sum_{i=1}^{p} h_i \frac{\partial}{\partial x_i}\left(\frac{\partial}{\partial x_j}f(\underset{\sim}{z}+t\underset{\sim}{h})\right)$. So

$$
\begin{aligned}
g''(t) &= \sum_{j=1}^{p} h_j \sum_{i=1}^{p} h_i \frac{\partial}{\partial x_i}\left(\frac{\partial}{\partial x_j}f(\underset{\sim}{z}+t\underset{\sim}{h})\right) && \text{(differentiate each term of } g'(t)\text{)} \\
&= \sum_{i=1}^{p}\sum_{j=1}^{p} h_i \left[\mathbf{H}_f(\underset{\sim}{z}+t\underset{\sim}{h})\right]_{ij} h_j && \text{(definition of the Hessian; reorder the finite sums)} \\
&= \underset{\sim}{h}^{\top}\mathbf{H}_f(\underset{\sim}{z}+t\underset{\sim}{h})\,\underset{\sim}{h}. && \text{(matrix product)}
\end{aligned}
$$

The last step is again the matrix product. Now apply Theorem 0.3 to $g$ on $[0,1]$: there is a $\tau \in (0,1)$ with

$$
\begin{aligned}
f(\underset{\sim}{z}+\underset{\sim}{h}) &= g(1) && \text{(definition of } g\text{)} \\
&= g(0) + g'(0)\,(1-0) + \frac{1}{2}\,g''(\tau)\,(1-0)^2 && \text{(Theorem 0.3, with } a=0, b=1\text{)} \\
&= g(0) + g'(0) + \frac{1}{2}\,g''(\tau) && (1-0=1) \\
&= f(\underset{\sim}{z}) + \underset{\sim}{h}^{\top}\frac{\partial}{\partial \underset{\sim}{x}}f(\underset{\sim}{z}) + \frac{1}{2}\,\underset{\sim}{h}^{\top}\mathbf{H}_f(\underset{\sim}{z}+\tau\underset{\sim}{h})\,\underset{\sim}{h} && \text{(substitute } g(0), g'(0), g''(\tau)\text{)} \\
&= f(\underset{\sim}{z}) + \left(\frac{\partial}{\partial \underset{\sim}{x}}f(\underset{\sim}{z})\right)^{\top}\underset{\sim}{h} + \frac{1}{2}\,\underset{\sim}{h}^{\top}\mathbf{H}_f(\underset{\sim}{z}+\tau\underset{\sim}{h})\,\underset{\sim}{h}. && \text{(a } 1\times 1 \text{ matrix equals its transpose)}
\end{aligned}
$$

□

**Example 0.6** (For a quadratic form the expansion is exact)**.** Let $f(\underset{\sim}{x}) = \underset{\sim}{x}^{\top}\mathbf{S}\underset{\sim}{x}$ with $\mathbf{S}$ symmetric and constant. Its gradient is $\nabla f(\underset{\sim}{x}) = 2\mathbf{S}\underset{\sim}{x}$ (quadratic form gradient) and its Hessian is $2\mathbf{S}$ at every point (quadratic form Hessian), so whatever $\tau$ is, Theorem 0.4 reads

$$
\begin{aligned}
f(\underset{\sim}{z}+\underset{\sim}{h}) &= \underset{\sim}{z}^{\top}\mathbf{S}\underset{\sim}{z} + (2\mathbf{S}\underset{\sim}{z})^{\top}\underset{\sim}{h} + \frac{1}{2}\,\underset{\sim}{h}^{\top}(2\mathbf{S})\underset{\sim}{h} \\
&= \underset{\sim}{z}^{\top}\mathbf{S}\underset{\sim}{z} + 2\,\underset{\sim}{z}^{\top}\mathbf{S}\underset{\sim}{h} + \underset{\sim}{h}^{\top}\mathbf{S}\underset{\sim}{h},
\end{aligned}
$$

using

$$
\begin{aligned}
(2\mathbf{S}\underset{\sim}{z})^\top &= 2\,\underset{\sim}{z}^\top \mathbf{S}^\top \\
&= 2\,\underset{\sim}{z}^\top \mathbf{S}.
\end{aligned}
$$

Multiplying out directly gives the same:

$$
\begin{aligned}
(\underset{\sim}{z}+\underset{\sim}{h})^\top \mathbf{S}(\underset{\sim}{z}+\underset{\sim}{h}) &= \underset{\sim}{z}^\top \mathbf{S}\underset{\sim}{z} + \underset{\sim}{z}^\top \mathbf{S}\underset{\sim}{h} + \underset{\sim}{h}^\top \mathbf{S}\underset{\sim}{z} + \underset{\sim}{h}^\top \mathbf{S}\underset{\sim}{h} && \text{(distribute)} \\
&= \underset{\sim}{z}^\top \mathbf{S}\underset{\sim}{z} + 2\,\underset{\sim}{z}^\top \mathbf{S}\underset{\sim}{h} + \underset{\sim}{h}^\top \mathbf{S}\underset{\sim}{h}. && (\underset{\sim}{h}^\top \mathbf{S}\underset{\sim}{z} = (\underset{\sim}{h}^\top \mathbf{S}\underset{\sim}{z})^\top, \text{ which equals } \underset{\sim}{z}^\top \mathbf{S}\underset{\sim}{h})
\end{aligned}
$$

**Definition 0.2** (Strict local minimizer)**.** Let $f : \mathbb{R}^p \to \mathbb{R}$. A point $\underset{\sim}{x}^*$ is a **strict local minimizer** of $f$ if there is a number $\delta > 0$ such that $f(\underset{\sim}{x}^*) < f(\underset{\sim}{x})$ for every $\underset{\sim}{x} \neq \underset{\sim}{x}^*$ with $\|\underset{\sim}{x} - \underset{\sim}{x}^*\| < \delta$.

**Example 0.7** (Strict and non-strict local minimizers)**.**

- $f(x) = (x-2)^2$ has

  $$
  \begin{aligned}
  f(2) &= 0 \\
  &< (x-2)^2 \\
  &= f(x)
  \end{aligned}
  $$

  for every $x \neq 2$, so 2 is a strict local minimizer (any $\delta > 0$ works).

- A constant function $f(\underset{\sim}{x}) = 0$ has $f(\underset{\sim}{x}^*) \leq f(\underset{\sim}{x})$ for all $\underset{\sim}{x}^*$ and $\underset{\sim}{x}$, so every point is a local minimizer, but no point is a strict one: $f(\underset{\sim}{x}^*) < f(\underset{\sim}{x})$ never holds.

**Lemma 0.1** (A quadratic form is at least the smallest eigenvalue times the squared length)**.** *Let* $\mathbf{A}$ *be a symmetric* $p \times p$ *matrix, and let* $\lambda_{\min}$ *be the smallest of its eigenvalues. Then* $\underset{\sim}{h}^\top \mathbf{A}\underset{\sim}{h} \geq \lambda_{\min} \|\underset{\sim}{h}\|^2$ *for every* $\underset{\sim}{h} \in \mathbb{R}^p$.

Proof

*Proof.* By the spectral theorem, write $\mathbf{A} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^\top$ with $\mathbf{Q}$ an orthogonal matrix and $\mathbf{\Lambda} = \operatorname{diag}(\lambda_1, \ldots, \lambda_p)$, and let $\underset{\sim}{y} = \mathbf{Q}^\top \underset{\sim}{h}$. Then

$$
\begin{aligned}
\underset{\sim}{h}^\top \mathbf{A} \underset{\sim}{h} &= \underset{\sim}{h}^\top \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^\top \underset{\sim}{h} && \text{(substitute the eigendecomposition)} \\
&= \underset{\sim}{y}^\top \mathbf{\Lambda} \underset{\sim}{y} && (\underset{\sim}{h}^\top \mathbf{Q} = (\mathbf{Q}^\top \underset{\sim}{h})^\top, \text{ transpose of a product}) \\
&= \sum_{i=1}^{p} \lambda_i \, y_i^2 && (\mathbf{\Lambda} \text{ is diagonal}) \\
&\geq \sum_{i=1}^{p} \lambda_{\min} \, y_i^2 && (\lambda_i \geq \lambda_{\min} \text{ and } y_i^2 \geq 0) \\
&= \lambda_{\min} \, \|\underset{\sim}{y}\|^2 && \text{(the squared length is the sum of squares)} \\
&= \lambda_{\min} \, \|\underset{\sim}{h}\|^2. && \text{(orthogonal matrices preserve length)}
\end{aligned}
$$

The second step is the transpose of a product. The last step applies orthogonal matrices preserve length to $\mathbf{Q}^\top$. That theorem applies because $\mathbf{Q}^\top$ is an orthogonal matrix: since $\mathbf{Q}$ is square with $\mathbf{Q}^\top\mathbf{Q} = \mathbf{I}_p$, also $\mathbf{Q}\mathbf{Q}^\top = \mathbf{I}_p$, so

$$(\mathbf{Q}^\top)^\top \mathbf{Q}^\top = \mathbf{Q}\mathbf{Q}^\top = \mathbf{I}_p.$$

□

**Example 0.8** (Checking the bound)**.** $\mathbf{A} = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$ has eigenvalues 3 and 1, so $\lambda_{\min} = 1$. At $\underset{\sim}{h} = (1, 0)^\top$, $\underset{\sim}{h}^\top \mathbf{A} \underset{\sim}{h} = 2 \geq 1 \cdot 1$. At $\underset{\sim}{h} = (1, -1)^\top$,

$$
\begin{aligned}
\underset{\sim}{h}^\top \mathbf{A} \underset{\sim}{h} &= 2 - 1 - 1 + 2 \\
&= 2 \\
&= 1 \cdot 2 :
\end{aligned}
$$

the bound holds with equality, since this $\underset{\sim}{h}$ is an eigenvector for $\lambda_{\min}$.

**Lemma 0.2** (A quadratic form with small entries is small)**.** *Let* $\mathbf{E}$ *be a* $p \times p$ *matrix whose entries all satisfy* $|e_{ij}| \leq m$*. Then* $|\underset{\sim}{h}^\top \mathbf{E} \underset{\sim}{h}| \leq p \, m \, \|\underset{\sim}{h}\|^2$ *for every* $\underset{\sim}{h} \in \mathbb{R}^p$*.*

Proof

*Proof.* Let $\underset{\sim}{a} = (|h_1|, \ldots, |h_p|)^\top$ and $\underset{\sim}{1} = (1, \ldots, 1)^\top$, both in $\mathbb{R}^p$. Then

$$
\begin{aligned}
\left|\underset{\sim}{h}^{\top}\mathbf{E}\underset{\sim}{h}\right| &= \left|\sum_{i=1}^{p}\sum_{j=1}^{p} h_i\, e_{ij}\, h_j\right| && \text{(matrix product)}\\
&\le \sum_{i=1}^{p}\sum_{j=1}^{p}|h_i|\,|e_{ij}|\,|h_j| && \text{(triangle inequality for numbers)}\\
&\le m\sum_{i=1}^{p}\sum_{j=1}^{p}|h_i|\,|h_j| && (|e_{ij}|\le m)\\
&= m\left(\sum_{i=1}^{p}|h_i|\right)^2 && \text{(the double sum factors)}\\
&= m\,(\underset{\sim}{1}\cdot\underset{\sim}{a})^2 && \text{(dot product)}\\
&\le m\,\|\underset{\sim}{1}\|^2\,\|\underset{\sim}{a}\|^2 && \text{(Cauchy-Schwarz, squared)}\\
&= m\,p\,\|\underset{\sim}{h}\|^2. && (\|\underset{\sim}{1}\|^2 = p \text{ and } \|\underset{\sim}{a}\|^2 = \textstyle\sum_i h_i^2, \text{ which equals } \|\underset{\sim}{h}\|^2)
\end{aligned}
$$

The steps use the matrix product, the dot product and the Cauchy-Schwarz inequality. □

**Example 0.9** (Checking the bound)**.** Let $\mathbf{E} = \begin{bmatrix} 0.1 & -0.2 \\ -0.2 & 0.1 \end{bmatrix}$, so $m = 0.2$ works. Let $\underset{\sim}{h} = (1,1)^{\top}$, so $\|\underset{\sim}{h}\|^2 = 2$. Then

$$
\begin{aligned}
\underset{\sim}{h}^{\top}\mathbf{E}\underset{\sim}{h} &= 0.1 - 0.2 - 0.2 + 0.1\\
&= -0.2,
\end{aligned}
$$

and

$$
\begin{aligned}
|-0.2| &= 0.2\\
&\le 2\cdot 0.2\cdot 2\\
&= 0.8.
\end{aligned}
$$

**Theorem 0.5** (Second-order sufficient condition)**.** *Let $f:\mathbb{R}^p\to\mathbb{R}$ have first and second partial derivatives that are continuous on $\mathbb{R}^p$. If $\underset{\sim}{x}^*$ is a stationary point of $f$ (Definition 0.1) and $\mathbf{H}_f(\underset{\sim}{x}^*)$ is positive definite, then $\underset{\sim}{x}^*$ is a strict local minimizer of $f$ (Definition 0.2).*

Proof

*Proof.* $\mathbf{H}_f(\underset{\sim}{x}^*)$ is positive definite, and so symmetric, so its smallest eigenvalue $\lambda$ is positive, by the eigenvalue test for definiteness. Each of the $p^2$ entries of $\mathbf{H}_f$ is continuous

at $\underset{\sim}{x}^*$: for $\varepsilon = \lambda/(2p)$ there is a $\delta_{ij} > 0$ with $\left|\left[\mathbf{H}_f(\underset{\sim}{y})\right]_{ij} - \left[\mathbf{H}_f(\underset{\sim}{x}^*)\right]_{ij}\right| < \lambda/(2p)$ whenever $\|\underset{\sim}{y} - \underset{\sim}{x}^*\| < \delta_{ij}$. Let $\delta$ be the smallest of these $p^2$ numbers.
Now let $0 < \|\underset{\sim}{h}\| < \delta$. By Theorem 0.4 there is a $\tau \in (0,1)$ for which the first step of the next display holds; let $\underset{\sim}{y} \overset{\text{def}}{=} \underset{\sim}{x}^* + \tau \underset{\sim}{h}$ and $\mathbf{E} \overset{\text{def}}{=} \mathbf{H}_f(\underset{\sim}{y}) - \mathbf{H}_f(\underset{\sim}{x}^*)$. Since $\|\underset{\sim}{y} - \underset{\sim}{x}^*\| = \tau\|\underset{\sim}{h}\| < \delta$, every entry of $\mathbf{E}$ has absolute value less than $\lambda/(2p)$, so Lemma 0.2 applies with $m = \lambda/(2p)$:

$$
\begin{aligned}
f(\underset{\sim}{x}^* + \underset{\sim}{h}) - f(\underset{\sim}{x}^*) &= \left(\frac{\partial}{\partial \underset{\sim}{x}} f(\underset{\sim}{x}^*)\right)^{\top} \underset{\sim}{h} + \frac{1}{2}\, \underset{\sim}{h}^{\top} \mathbf{H}_f(\underset{\sim}{y})\, \underset{\sim}{h} && \text{(Theorem 0.4)} \\
&= \frac{1}{2}\, \underset{\sim}{h}^{\top} \mathbf{H}_f(\underset{\sim}{y})\, \underset{\sim}{h} && (\underset{\sim}{x}^* \text{ is stationary}) \\
&= \frac{1}{2}\, \underset{\sim}{h}^{\top} \mathbf{H}_f(\underset{\sim}{x}^*)\, \underset{\sim}{h} + \frac{1}{2}\, \underset{\sim}{h}^{\top} \mathbf{E}\, \underset{\sim}{h} && (\mathbf{H}_f(\underset{\sim}{y}) = \mathbf{H}_f(\underset{\sim}{x}^*) + \mathbf{E}\text{; distribute}) \\
&\geq \frac{1}{2}\, \lambda\, \|\underset{\sim}{h}\|^2 + \frac{1}{2}\, \underset{\sim}{h}^{\top} \mathbf{E}\, \underset{\sim}{h} && \text{(Lemma 0.1)} \\
&\geq \frac{1}{2}\, \lambda\, \|\underset{\sim}{h}\|^2 - \frac{1}{2}\, \left|\underset{\sim}{h}^{\top} \mathbf{E}\, \underset{\sim}{h}\right| && \text{(a number is at least minus its absolute value)} \\
&\geq \frac{1}{2}\, \lambda\, \|\underset{\sim}{h}\|^2 - \frac{1}{2}\, p\, \frac{\lambda}{2p}\, \|\underset{\sim}{h}\|^2 && \text{(Lemma 0.2, } m = \lambda/(2p)) \\
&= \frac{\lambda}{4}\, \|\underset{\sim}{h}\|^2 && \text{(arithmetic)} \\
&> 0. && (\lambda > 0 \text{ and } \underset{\sim}{h} \neq \underset{\sim}{0})
\end{aligned}
$$

So $f(\underset{\sim}{x}^*) < f(\underset{\sim}{x})$ whenever $0 < \|\underset{\sim}{x} - \underset{\sim}{x}^*\| < \delta$ (take $\underset{\sim}{h} = \underset{\sim}{x} - \underset{\sim}{x}^*$). □

**Example 0.10** (Classifying stationary points)**.**

- $f(\underset{\sim}{x}) = x_1^2 + x_2^2 - 2x_1$ has the stationary point $(1,0)^{\top}$ (Example 0.2), and its Hessian is

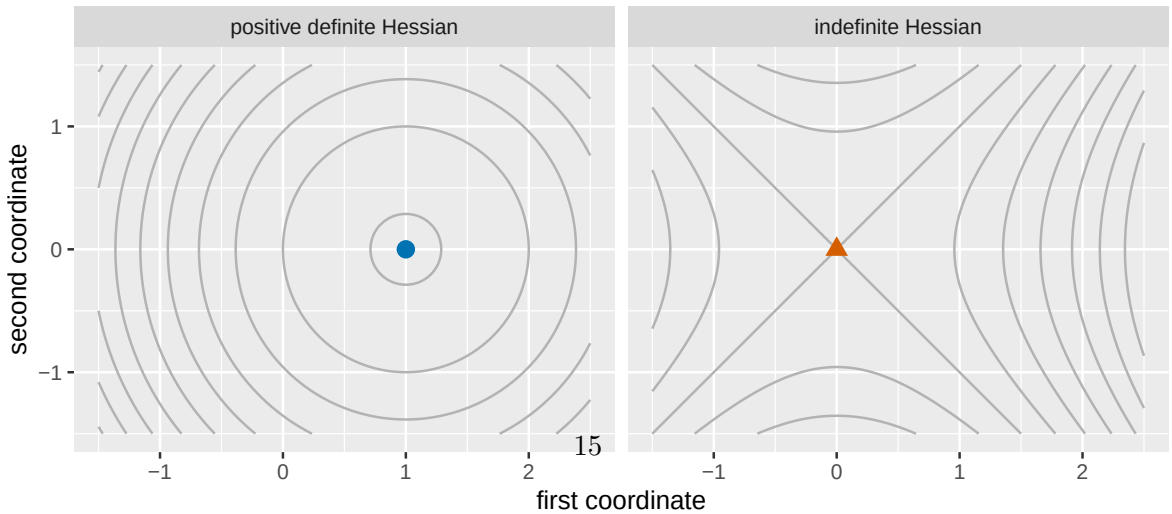
$$
\begin{aligned}
\mathbf{H}_f(\underset{\sim}{x}) &= \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \\
&= 2\mathbf{I}_2,
\end{aligned}
$$

  which is positive definite, so $(1,0)^{\top}$ is a strict local minimizer.

- $f(\underset{\sim}{x}) = x_1^2 - x_2^2$ has the stationary point $\underset{\sim}{0}$ and Hessian $\begin{bmatrix} 2 & 0 \\ 0 & -2 \end{bmatrix}$, which is not positive definite ($(0,1)^{\top}$ gives $-2$). Indeed, $\underset{\sim}{0}$ is not a local minimizer (Example 0.3).

### R

```
so_x_so <- seq(-1.5, 2.5, length.out = 101)
so_y_so <- seq(-1.5, 1.5, length.out = 101)
so_grid_so <- tidyr::expand_grid(x = so_x_so, y = so_y_so)
so_gx_so <- so_grid_so$x
so_gy_so <- so_grid_so$y
so_bowl_so <- so_gx_so^2 + so_gy_so^2 - 2 * so_gx_so
so_saddle_so <- so_gx_so^2 - so_gy_so^2
so_bowl_tbl_so <- dplyr::mutate(
  so_grid_so, z = so_bowl_so, panel = "positive definite Hessian"
)
so_saddle_tbl_so <- dplyr::mutate(
  so_grid_so, z = so_saddle_so, panel = "indefinite Hessian"
)
so_all_so <- dplyr::bind_rows(so_bowl_tbl_so, so_saddle_tbl_so)
so_order_so <- c("positive definite Hessian", "indefinite Hessian")
so_all_so$panel <- factor(so_all_so$panel, levels = so_order_so)
so_min_panel_so <- factor(so_order_so[1], levels = so_order_so)
so_sad_panel_so <- factor(so_order_so[2], levels = so_order_so)
so_min_so <- tibble::tibble(x = 1, y = 0, panel = so_min_panel_so)
so_sad_so <- tibble::tibble(x = 0, y = 0, panel = so_sad_panel_so)
so_aes_so <- ggplot2::aes(x, y)
ggplot2::ggplot(so_all_so, so_aes_so) +
  ggplot2::geom_contour(ggplot2::aes(z = z), colour = "grey70", bins = 12) +
  ggplot2::geom_point(data = so_min_so, colour = "#0072B2", size = 3) +
  ggplot2::geom_point(
    data = so_sad_so, colour = "#D55E00", size = 3, shape = 17
  ) +
  ggplot2::facet_wrap(~panel) +
  ggplot2::coord_equal() +
  ggplot2::labs(x = "first coordinate", y = "second coordinate")
```






### Python

```
import matplotlib.pyplot as plt
```

**R**

*(This 3D surface is interactive; see the HTML or slide version of this page to view it.)*

**Python**

```
import matplotlib.pyplot as plt
import numpy as np

so_x_3d = np.linspace(-1.5, 2.5, 101)
so_y_3d = np.linspace(-1.5, 1.5, 101)
so_xx_3d, so_yy_3d = np.meshgrid(so_x_3d, so_y_3d)
so_bowl_3d = so_xx_3d**2 + so_yy_3d**2 - 2 * so_xx_3d
so_saddle_3d = so_xx_3d**2 - so_yy_3d**2
fig = plt.figure(figsize=(7, 3.5))
ax_bowl = fig.add_subplot(1, 2, 1, projection="3d")
ax_bowl.plot_surface(so_xx_3d, so_yy_3d, so_bowl_3d, color="#0072B2", alpha=0.6)
ax_bowl.scatter([1], [0], [-1], marker="o", color="#0072B2", s=50)
ax_bowl.set_title("positive definite Hessian")
ax_saddle = fig.add_subplot(1, 2, 2, projection="3d")
ax_saddle.plot_surface(so_xx_3d, so_yy_3d, so_saddle_3d, color="#0072B2",
                       alpha=0.6)
ax_saddle.scatter([0], [0], [0], marker="D", color="#D55E00", s=50)
ax_saddle.set_title("indefinite Hessian")
for ax in (ax_bowl, ax_saddle):
    ax.set_xlabel("x")
    ax.set_ylabel("y")
    ax.set_zlabel("f")
plt.show();
```

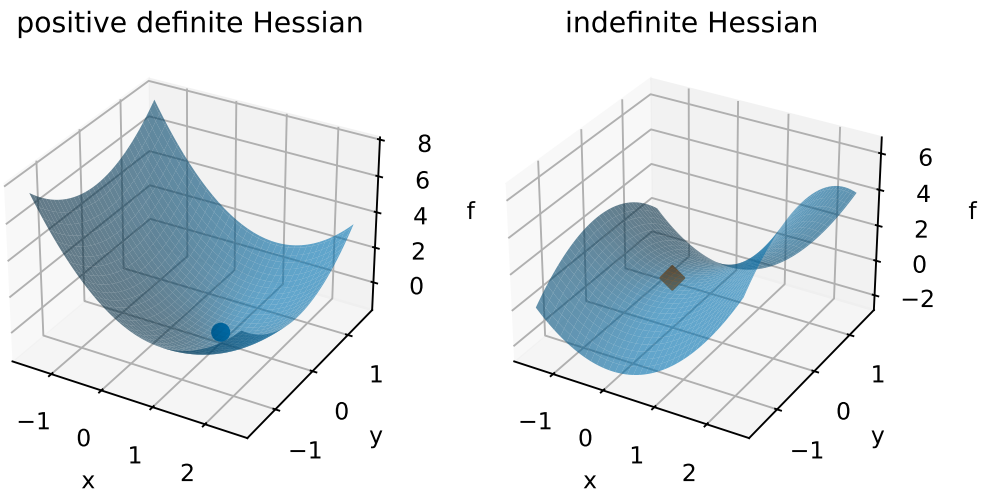





Figure 4: The graphs of the two functions in Example 0.10, the surfaces whose level curves are in Figure 3. The surface of $x_1^2 + x_2^2 - 2x_1$ is a bowl, and its stationary point $(1, 0)^\top$ (circle) is the bottom, a strict local minimizer. The surface of $x_1^2 - x_2^2$ is a saddle: it rises along the first coordinate axis and falls along the second, so the stationary point $\underset{\sim}{0}$ (diamond) is not a local minimizer.

**Example 0.11** (The condition is sufficient, not necessary)**.** $f(\underset{\sim}{x}) = x_1^2 + x_2^4$ and $g(\underset{\sim}{x}) = x_1^2 - x_2^4$ both have gradient $\underset{\sim}{0}$ at $\underset{\sim}{0}$ and the same Hessian there, $\begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix}$, since

$$\begin{aligned} \frac{\partial}{\partial x_2}\left(\frac{\partial}{\partial x_2}(\pm x_2^4)\right) &= \pm 12 x_2^2 \\ &= 0 \end{aligned}$$

at $x_2 = 0$. That Hessian is not positive definite ($(0,1)^\top$ gives 0), so Theorem 0.5 says nothing about either function. In fact $\underset{\sim}{0}$ is a strict local minimizer of $f$, since $f(\underset{\sim}{x}) > 0 = f(\underset{\sim}{0})$ for $\underset{\sim}{x} \neq \underset{\sim}{0}$, but not a local minimizer of $g$, since

$$\begin{aligned} g(0,t) &= -t^4 \\ &< 0 \\ &= g(\underset{\sim}{0}) \end{aligned}$$

for $t \neq 0$.

**Definition 0.3** (Local maximizer)**.** Let $f : \mathbb{R}^p \to \mathbb{R}$. A point $\underset{\sim}{x}^* \in \mathbb{R}^p$ is a **local maximizer** of $f$ if there is a number $\delta > 0$ such that $f(\underset{\sim}{x}^*) \geq f(\underset{\sim}{x})$ for all $\underset{\sim}{x} \in \mathbb{R}^p$ with $\|\underset{\sim}{x} - \underset{\sim}{x}^*\| < \delta$.

**Example 0.12** (A local maximizer)**.** Let $f(x) = 1 - x^2$. For every $x \in \mathbb{R}$,

$$\begin{aligned} f(0) - f(x) &= 1 - (1 - x^2) && \text{(substitute)} \\ &= x^2 && \text{(simplify)} \\ &\geq 0, && \text{(a square)} \end{aligned}$$

so $f(0) \geq f(x)$ for all $x$, and 0 is a local maximizer of $f$ with any $\delta > 0$.

**Definition 0.4** (Saddle point)**.** Let $f : \mathbb{R}^p \to \mathbb{R}$. A stationary point of $f$ that is neither a local minimizer nor a local maximizer of $f$ is a **saddle point** of $f$ [5].

**Example 0.13** (A saddle point)**.** Let $f(\underset{\sim}{x}) = x_1^2 - x_2^2$. Its gradient is

$$\nabla f(\underset{\sim}{x}) = (2x_1,\ -2x_2)^\top.$$

At $\underset{\sim}{x} = \underset{\sim}{0}_{2\times 1}$, the gradient is $\nabla f(\underset{\sim}{0}_{2\times 1}) = \underset{\sim}{0}_{2\times 1}$, so $\underset{\sim}{0}_{2\times 1}$ is a stationary point.
For every $t \neq 0$,

$$
\begin{aligned}
f(t,0) &= t^2 && \text{(substitute)} \\
&> 0 && (t \neq 0) \\
&= f(\underset{\sim}{0}_{2\times 1}), && \text{(substitute)} \\
f(0,t) &= -t^2 && \text{(substitute)} \\
&< 0 && (t \neq 0) \\
&= f(\underset{\sim}{0}_{2\times 1}). && \text{(substitute)}
\end{aligned}
$$

So every ball around $\underset{\sim}{0}_{2\times 1}$ contains a point with a larger value of $f$ and a point with a smaller one. Hence $\underset{\sim}{0}_{2\times 1}$ is neither a local minimizer nor a local maximizer, and it is a saddle point. Figure 5 shows the surface: the curve $t \mapsto f(t,0)$ rises from the origin along the $x_1$ axis, and the curve $t \mapsto f(0,t)$ falls from the origin along the $x_2$ axis.

**R**

*(This 3D surface is interactive; see the HTML or slide version of this page to view it.)*

**Python**

```
import matplotlib.pyplot as plt
import numpy as np

sp_t_3d = np.linspace(-1, 1, 61)
sp_x1_3d, sp_x2_3d = np.meshgrid(sp_t_3d, sp_t_3d)
sp_z_3d = sp_x1_3d**2 - sp_x2_3d**2
sp_zero_3d = np.zeros_like(sp_t_3d)
fig = plt.figure(figsize=(6, 4.5))
ax = fig.add_subplot(projection="3d")
ax.plot_surface(sp_x1_3d, sp_x2_3d, sp_z_3d, color="#0072B2", alpha=0.6)
ax.plot(sp_t_3d, sp_zero_3d, sp_t_3d**2, color="#D55E00",
        label="f(t, 0) = t^2")
ax.plot(sp_zero_3d, sp_t_3d, -sp_t_3d**2, "--", color="#D55E00",
        label="f(0, t) = -t^2")
ax.scatter([0], [0], [0], color="black", label="saddle point")
ax.set_xlabel("x1")
ax.set_ylabel("x2")
ax.set_zlabel("f(x)")
ax.legend(loc="upper right")
plt.show();
```

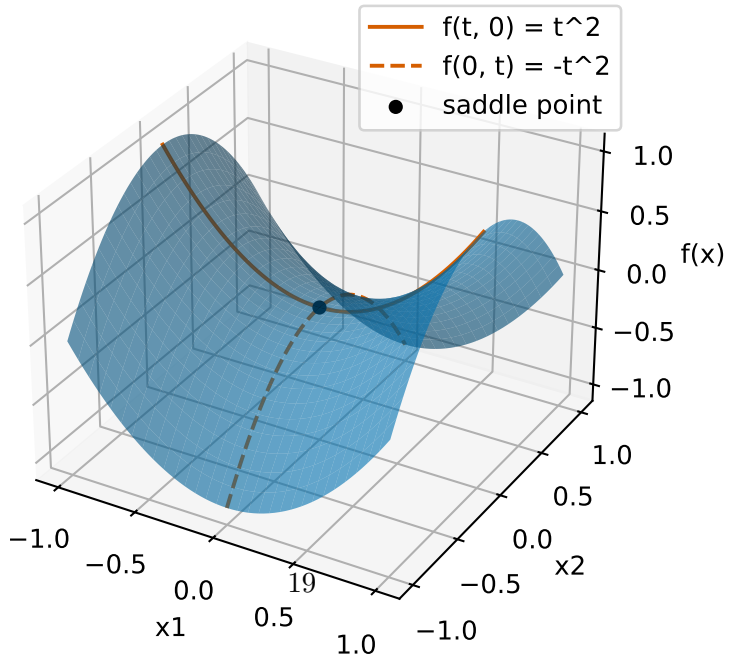



Figure 5: The graph of $f(\underset{\sim}{x}) = x_1^2 - x_2^2$. Along the $x_1$ axis (solid curve) the surface rises away from the origin, and along the $x_2$ axis (dashed curve) it falls away, so the

[1] H. Zhou, "Optimization and multivariate calculus." Lecture notes for Biostat 216, Mathematical Methods for Biostatistics, University of California, Los Angeles, 2024. Accessed: Oct. 05, 2026. [Online]. Available: https://ucla-biostat-216.github.io/2024fall/slides/13-optim/13-optim.html

[2] Wikipedia contributors, "Fermat's theorem (stationary points) — Wikipedia, the free encyclopedia." 2026. Available: https://en.wikipedia.org/wiki/Fermat%27s_theorem_(stationary_points)

[3] Wikipedia contributors, "Taylor's theorem — Wikipedia, the free encyclopedia." 2026. Available: https://en.wikipedia.org/wiki/Taylor%27s_theorem

[4] W. Rudin, *Principles of mathematical analysis*, 3rd ed. in International series in pure and applied mathematics. New York: McGraw-Hill, 1976.

[5] Wikipedia contributors, "Saddle point — Wikipedia, the free encyclopedia." 2026. Available: https://en.wikipedia.org/wiki/Saddle_point