

# Binary outcomes: who survived the Titanic?

Programming & Data Management, Part II · Further topic F2  
(optional, not examined)

Magnus Lodefalk  
*Örebro University*

NA446A, Örebro University  
Autumn 2026



# Today's goals

By the end of today you can:

1. Estimate a **linear probability model** and a **logit** on a 0/1 outcome with `statsmodels`
2. Read the logit three ways: **log-odds, odds ratios and average marginal effects**, and know which one answers “how many percentage points?”
3. Spot **perfect separation** before it spoils a model, and fix it

# Six terms for this module

---

|                          |                                                                                    |
|--------------------------|------------------------------------------------------------------------------------|
| linear probability model | OLS on a 0/1 outcome; a coefficient is a change in the probability                 |
| logit                    | a model that keeps the probability between 0 and 1:<br>$p = 1 / (1 + e^{-x\beta})$ |
| odds                     | $p / (1 - p)$ : a survival probability of 0.75 is odds of 3                        |
| odds ratio               | $e^{\beta}$ : the factor by which the <i>odds</i> change                           |
| average marginal effect  | the change in the probability, averaged over the people in the data                |
| perfect separation       | a category in which everyone has the same outcome                                  |

---

- Nothing to install: `statsmodels` is in the course environment. The data are Session 2's `titanic.csv`.

## Two models for a 0/1 outcome

---

## Step 1: the data, and one stated choice

```
t = pd.read_csv(HERE / "data" / "titanic.csv")
t.columns = t.columns.str.lower()
t["female"] = (t["sex"] == "female").astype(int)
d = t.dropna(subset=["age"])
print(len(t), len(d), round(d["survived"].mean(), 3))
```

```
| 891 714 0.406
```

177 passengers have no recorded age. The models use the 714 whose age is known, of whom 40.6 per cent survived.

- Dropping rows is a choice: say it, count it, and ask whether those passengers differ (Session 2's intake checklist).

Stata: `import delimited titanic.csv, case(lower); gen female = sex == "female".` Stata drops the missing ages silently inside each model.

## Step 2: the linear probability model

```
lpm = smf.ols(f"survived ~ {RHS}", data=d).fit(  
    cov_type="HC1")  
p_hat = lpm.fittedvalues  
print((p_hat < 0).sum(), (p_hat > 1).sum())
```

```
female          0.479  
C(pclass)[T.3]  -0.407  
21 20
```

RHS = "female + C(pclass) + age"; C(): class dummies, first class the base.

- Women 48 percentage points more likely to survive; but 41 of 714 predictions lie outside 0 and 1.

Stata: reg survived i.female i.pclass age, vce(robust): same numbers. HC1, since a 0/1 outcome is heteroskedastic by construction.

## Steps 3 and 4: the logit, in log-odds and as odds ratios

```
logit = smf.logit(f"survived ~ {RHS}", data=d).fit(
    disp=False)
odds_ratio = np.exp(logit.params)
```

|                 | coef   | se    | odds ratio |
|-----------------|--------|-------|------------|
| Intercept       | 1.254  | 0.360 | 3.51       |
| C(pclass) [T.2] | -1.310 | 0.278 | 0.27       |
| C(pclass) [T.3] | -2.581 | 0.281 | 0.08       |
| female          | 2.523  | 0.207 | 12.46      |
| age             | -0.037 | 0.008 | 0.96       |

- 2.52 is a change in the *log*-odds;  $e^{2.52} = 12.46$ : a woman's odds of survival were 12.5 times a man's, class and age held fixed.

Stata: `logit survived i.female i.pclass age, then logit, or;` same numbers.

## Check your understanding

Answer on your own first, then compare with a neighbour

A news story says: “The odds ratio is 12.5, so women were 12.5 times as likely to survive.” Predict: roughly how many times as likely were they, in these data?

## Check your understanding

Answer on your own first, then compare with a neighbour

A news story says: “The odds ratio is 12.5, so women were 12.5 times as likely to survive.” Predict: roughly how many times as likely were they, in these data?

**Answer:** About 3.7 times: 75.5 per cent of the women survived and 20.5 per cent of the men. The odds ratio compares *odds*:  $0.755/0.245 = 3.1$  against  $0.205/0.795 = 0.26$ , a ratio of 11.9 (12.5 with class and age held fixed). Odds and probabilities are close only when the probability is small. For “how many percentage points”, use the marginal effect.

# Marginal effects, and a warning

---

## Step 5: the average marginal effect, by hand

```
p_woman = logit.predict(d.assign(female=1))  
p_man = logit.predict(d.assign(female=0))  
ame_female = (p_woman - p_man).mean()
```

```
female: 0.474 | third vs first class: -0.424
```

Every passenger once as a woman and once as a man, class and age as they are; the average difference in the predicted probability. Class: everyone in third, then everyone in first.

- 47 percentage points: close to the LPM's 48. Here the two agree on the average; the LPM goes wrong for passengers near 0 and 1.

Stata: margins, dydx(\*) after logit: 0.474 and -0.424.

## The same with `get_margeff`, and two traps

```
ame = logit.get_margeff(at="overall", dummy=True)
```

|                            | LPM    | get_margeff | by hand |
|----------------------------|--------|-------------|---------|
| female                     | 0.479  | 0.474       | 0.474   |
| C(pclass)[T.3]             | -0.407 | -0.409      | -0.424  |
| female without dummy=True: | 0.367  |             |         |

- Without `dummy=True`, `statsmodels` takes the *slope* at each point (0.367), the wrong question for a 0/1 variable
- For a category with three levels, it switches one dummy on without switching the other off (a second-class passenger gets “third class” as well):  $-0.409$ , where Stata's `margins` and the by-hand version give  $-0.424$
- For dummies and categories, the three lines by hand are the safe route.

## Step 6: perfect separation

```
pd.crosstab(t["sibsp"], t["survived"]).T
```

| sibsp    | 0   | 1   | 2  | 3  | 4  | 5 | 8 |
|----------|-----|-----|----|----|----|---|---|
| survived |     |     |    |    |    |   |   |
| 0        | 398 | 97  | 15 | 12 | 15 | 5 | 7 |
| 1        | 210 | 112 | 13 | 4  | 3  | 0 | 0 |

Nobody with 5 or 8 siblings or spouses aboard survived. A logit with `C(sibsp)`:

```
warning: Maximum Likelihood optimization failed to converge  
              coef      se  
C(sibsp)[T.5] -20.4  13752.9
```

- -20 with a standard error of 13,753: the optimiser chased minus infinity and gave up. Only a warning: the script runs on.

Stata stops you instead: "5.sibsp != 0 predicts failure perfectly", and drops those 12 passengers.

## Step 7: the fix, merge the rare categories

```
t["sibsp_cap"] = t["sibsp"].clip(upper=CAP)
fixed = smf.logit("survived ~ female + C(sibsp_cap)",
                  data=t).fit(dispatch=False)
```

```
converged: True
```

|                    | coef  | se   |
|--------------------|-------|------|
| C(sibsp_cap) [T.3] | -1.96 | 0.46 |

- CAP = 3: 3, 4, 5 and 8 become “3 or more”, which has survivors. Tabulate the outcome against every categorical regressor first.

Stata: `gen sibsp_cap = min(sibsp, 3), logit survived i.female i.sibsp_cap: -1.96 (0.46)`. With person dummies in a panel logit, use `xtlogit`, `fe` or `statsmodels' ConditionalLogit`.

# Summary, and where to go next

1. `smf.ols(...).fit(cov_type="HC1")` for the LPM; `smf.logit(...)` for the logit
2. Log-odds, odds ratios (`np.exp`) and marginal effects answer different questions; percentage points come from the AME, by hand for dummies
3. Cross-tabulate before you estimate: an empty cell breaks a logit quietly

## Read next (open, checked 30 Sep 2026):

- statsmodels, *Discrete choice models overview* (logit, probit, marginal effects):  
[statsmodels.org/stable/examples/notebooks/generated/discrete\\_choice\\_overview.html](https://statsmodels.org/stable/examples/notebooks/generated/discrete_choice_overview.html)
- statsmodels, *Regression with discrete dependent variable* (the reference):  
[statsmodels.org/stable/discretemod.html](https://statsmodels.org/stable/discretemod.html)