MedGemma ■■■■■■■■

Python ■■■■■■■■■■

Notebook 20 ■ Cells ■■■■

■■■■■■■■■■■■■■■■■■■■■

Cell 1: ■■■■

■■■■■

```
print('■ ■■■■■■■■...')
!pip install -q transformers datasets accelerate bitsandbytes peft openpyxl scikit-learn matplo
print('■ ■■■■■■■')
```

■■■■■

■■ Cell ■■■■■■■■■ Python ■■■

• ! ■■■■ Jupyter/Colab ■■ Shell ■■

• pip■Python ■■■■■■

• -q■■■■■■■■■■

■■■■

1. transformers■HuggingFace ■■■

2. datasets■■■■■■

3. bitsandbytes■■■■■■4-bit■

4. peft■LoRA ■■

5. scikit-learn■■■■■

■■■■

■■■■■ 32-bit ■■■■■ 4-bit■■■ 8 ■■■■■

■■■■■■■■■■■■■■■■■■■■■■■■■■■■

Cell 9: [redacted]

[redacted]

[redacted] 900 [redacted] 100 [redacted] 90%[redacted]

```
TARGET_ERROR_RATIO = 0.40  # [redacted] 40%
MAX_TOTAL_SAMPLES = 3500   # [redacted]

# [redacted]
df_error_upsampled = resample(df_error, replace=True, n_samples=needed_count)

# [redacted]
df = pd.concat([df_correct, df_error_upsampled])
df = df.sample(frac=1, random_state=42).reset_index(drop=True)
```

[redacted]

- [redacted]replace=True[redacted]
- [redacted]replace=False[redacted]
- random_state=42[redacted]

Cell 17: ■■■■

```
values = [0.85, 0.87, 0.83, 0.86, 0.84]

mean = np.mean(values)  # 0.85
std = np.std(values, ddof=1)  # 0.0158

# 95% ■■■■
ci = stats.t.interval(0.95, df=4, loc=mean, scale=stats.sem(values))
```

■■■■

■■■■■■■■■■5 ■■■■■■■■

■■■■■■■■■■■■■■■■■■

■■■■■■■■■■ 95% ■■■■■■

■■■■■■■■

20 ■ Cells ■■■■

Cell 1-2■■■■■■■■■■■■■

Cell 3-5■■■■■■GPU■Drive■■■■

Cell 6-8■■■■■■■■

Cell 9■■■■■

Cell 10-11■■■■■

Cell 12-16■■■■■■5-fold■

Cell 17■■■■■

Cell 18-19■■■■

Cell 20■■■■■

■■■■

■■■■32-bit → 4-bit■■■ 8 ■■■■

■LoRA■■■■ 0.21% ■■

■■■■■■5-fold■■■■■■

■■■■■■■■■ MedGemma ■■■■■