Module 4:

Team Members:

Logan Heselton, Jaya Kinley

Project Title:

Modeling and Forecasting MERS Weekly Case Counts (2013–2014)

Project Goal:

This project seeks to analyze weekly MERS case data from 2013–2014 and develop a predictive model capable of forecasting future case counts.

Disease Background:

- Prevalence & incidence
    - Incidence: A total of 425 cases were recorded in Saudi Arabia between June 6, 2013 and May 14, 2014, with the highest number of cases and deaths occurring between April and May 2014 (Alghamdi et al., 2014).
        - Weekly incidence: ~8.7 new MERS cases per week
    - Clinical course: “The median time from symptom onset to hospitalization was 4 days, and to death was 12 days.” (Assiri et al., 2013)
        - Because MERS has a short clinical course, the number of individuals simultaneously ill at any given time is likely low, indicating low point prevalence during the outbreak.

    Sources:

    Alghamdi, I. G., Hussain, I. I., Alghamdi, M. S., Alghamdi, M. M., & El-Sheemy, M. A. (2014). The pattern of Middle East respiratory syndrome coronavirus in Saudi Arabia: An epidemiological analysis of data from the Saudi Ministry of

Health. *International Journal of Infectious Diseases, 29*, 10–22. https://doi.org/10.1016/j.ijid.2014.05.007

Assiri, A., McGeer, A., Perl, T. M., Price, C. S., Al Rabeeah, A. A., Cummings, D. A., … KSA MERS-CoV Investigation Team. (2013). Hospital outbreak of Middle East respiratory syndrome coronavirus. *New England Journal of Medicine, 369*(5), 407–416. https://doi.org/10.1056/NEJMoa1306742

- Economic burden
    - Serious impact on public health and safety because it overtaxed the healthcare system, disrupted social and economic activity and the regions of outbreak, and disrupted travel between Saudi Arabia and other neighboring regions.
    - Significant strain on healthcare systems and significant morbidity and mortality in those affected.
    - The total cost of managing a MERS case at the hospital ranged from $1278.41 to $75,987.95 with a mean cost of $12,947.03 ± $19,923.14.

      Sources:

      Salomon I. Saudi Arabia's Middle East respiratory syndrome Coronavirus (MERS-CoV) outbreak: consequences, reactions, and takeaways. Ann Med Surg (Lond). 2024 Jul 1;86(8):4668-4674. doi: 10.1097/MS9.0000000000002336. PMID: 39118758; PMCID: PMC11305771.

      AlRuthia Y, Somily AM, Alkhamali AS, Bahari OH, AlJuhani RJ, Alsenaidy M, Balkhi B. Estimation Of Direct Medical Costs Of Middle East Respiratory Syndrome Coronavirus Infection: A Single-Center Retrospective Chart Review Study. Infect Drug Resist. 2019 Nov 7;12:3463-3473. doi: 10.2147/IDR.S231087. PMID: 31819541; PMCID: PMC6844224.
- Risk factors (genetic, lifestyle) & Societal determinants
    - Substantial risk factors among camel workers in Qatar: involvement in animal training, milking camels, workers

with respiratory symptoms requiring an overnight stay in hospital, contact with camels' waste, poor hand hygeiene before and after animal tasks.

- Independent risk factors of primary MERS-CoV infection in Saudi Arabia: direct dromedary exposure in the 2 weeks before illness onset/direct physical contact with dromedary camels during the previous 6 months, diabetes, heart disease, currently smoking tobacco.

  Source:

  Hui DS, Azhar EI, Kim YJ, Memish ZA, Oh MD, Zumla A. Middle East respiratory syndrome coronavirus: risk factors and determinants of primary, household, and nosocomial transmission. Lancet Infect Dis. 2018 Aug;18(8):e217-e227. doi: 10.1016/S1473-3099(18)30127-0. Epub 2018 Apr 18. PMID: 29680581; PMCID: PMC7164784.

- Symptoms
  - The symptoms for MERS-CoV range from asymptomatic or mild respiratory symptoms to severe acute respiratory disease. In some cases, some patients are asymptomatic, but most develop symptoms after 5 days of exposure and can even first appear up to 14 days after. Typical symptoms include fever, cough, shortness of breath, muscle aches, sore throat, runny nose, and chills. Other symptoms, however uncommon, include bloody coughing and gastrointestinal symptoms, such as diarrhea and vomiting. Severe symptoms are more common in older adults or people who have preexisting chronic illnesses, such as diabetes, cancer, or lung disease, and a weakened immune system. These complications include phenomena, failure of the kidneys or other organs, septic shock, and respiratory failure, which requires the need for a ventilator.
  - Source: https://www.hopkinsmedicine.org/health/conditions-and-diseases/middle-east-respiratory-syndrome-mers
- Diagnosis
  - Molecular tests are most commonly used to diagnose MERS-CoV. NAATs such as real-time reverse-transcription

polymerase chain reaction (rRT-PCR) assays detect viral genetic material in clinical specimens. These assays target specific regions of the MERS-CoV genome, most often the upE (upstream of the E gene), ORF1a, and ORF1b regions, to confirm infection with high sensitivity and specificity. Diagnostic testing typically requires respiratory specimens, with lower respiratory tract samples (such as sputum, bronchoalveolar lavage, or tracheal aspirates) providing the highest viral yield. In cases where molecular tests are negative but clinical suspicion of MERS-CoV remains high, repeat testing with additional specimen types or later in the course of illness is recommended, because viral load can vary by sample site and time after symptom onset. Serologic assays may be used for retrospective diagnosis or surveillance, but they are not the primary tools for acute clinical diagnosis.

        - Source: https://www.cdc.gov/mers/php/laboratories/index.html

    - Other diagnostic approaches for MERS-CoV include serologic tests, such as ELISA, indirect fluorescent antibody assays, and microneutralization tests, which detect antibodies produced later in infection and are mainly used for retrospective confirmation rather than early diagnosis. Viral culture can also identify MERS-CoV, but it requires high-level biosafety containment and is therefore limited to specialized reference laboratories. Genomic sequencing of viral material may be performed to confirm infection and track viral evolution, though it is not a frontline clinical test. Additionally, radiologic imaging, such as chest X-rays or CT scans, can reveal pneumonia patterns that support the diagnosis, but imaging alone cannot confirm MERS-CoV.

        - Source: https://archive.cdc.gov/www_cdc_gov/coronavirus/mers/guidelines-clinical-specimens.html?utm_source=chatgpt.com

- Biological mechanisms (anatomy, organ physiology, cell & molecular physiology)

    - Anatomy

        - MERS-CoV specifically targets the respiratory tract, beginning in the nasal passages and upper

airways before moving into the lower respiratory system. The virus most efficiently infects the bronchi, bronchioles, and especially the alveoli, where the DPP4 receptor is highly expressed on epithelial cells. Interestingly, the DPP4 gene is highly linked to both bat-CoVs (HKU4) and MERS-CoV, as both have the same receptor in cell tropism features. This targeting of the pulmonary capillary network and the alveoli leads to the infection rapidly triggering inflammation, fluid leakage, and impaired gas exchange. In severe cases, MERS-CoV can extend beyond the lungs, affecting organs such as the kidneys and gastrointestinal tract.
    - Source: https://pmc.ncbi.nlm.nih.gov/articles/PMC7409282/
- Organ Physiology
    - At the organ level, MERS-CoV disrupts the normal physiology of the lungs, which are responsible for gas exchange. Infection leads to alveolar inflammation, edema, and impaired oxygen transfer, producing symptoms like shortness of breath and hypoxemia. The respiratory system's physiological response includes coughing, mucus production, and immune-driven inflammation can contribute to respiratory failure or acute respiratory distress syndrome (ARDS). Kidney physiology may also be affected; MERS-CoV has been detected in renal tissue, and some patients develop acute kidney injury due to direct viral effects or systemic inflammatory responses.
    - Source: https://pmc.ncbi.nlm.nih.gov/articles/PMC7409282/
- Cell and Molecular Physiology
    - At the cell and molecular level, MERS-CoV infects host cells by binding its spike (S) protein to the DPP4 receptor on epithelial cells of the airway and alveoli. After entry, the virus releases its RNA genome, hijacking the host cell's

machinery to synthesize viral proteins and replicate new virions. This replication triggers innate immune pathways, including interferon responses, but MERS-CoV employs several molecular strategies to suppress interferon signaling, allowing uncontrolled viral replication. Infected cells undergo dysfunction and apoptosis, releasing inflammatory cytokines that amplify tissue damage.
    - Source: https://www.nature.com/articles/s44298-024-00080-y

Dataset:

The dataset analyzed in this project came from the MERS Outbreak Dataset (2012-2019), available on Kaggle (Imdevskp, 2020). The file weekly_clean.csv provides weekly counts of laboratory-confirmed MERS-CoV cases. Each row represents a single epidemiological week. The file as two variables: date and confirmed_cases. The original data was obtained from public reports provided by the World Health Organization (WHO) outbreak reports and the Saudi Arabian Ministry of Health (MOH) MERS-CoV situation updates.

Source: https://www.kaggle.com/datasets/imdevskp/mers-outbreak-dataset-20122019?select=weekly_clean.csv

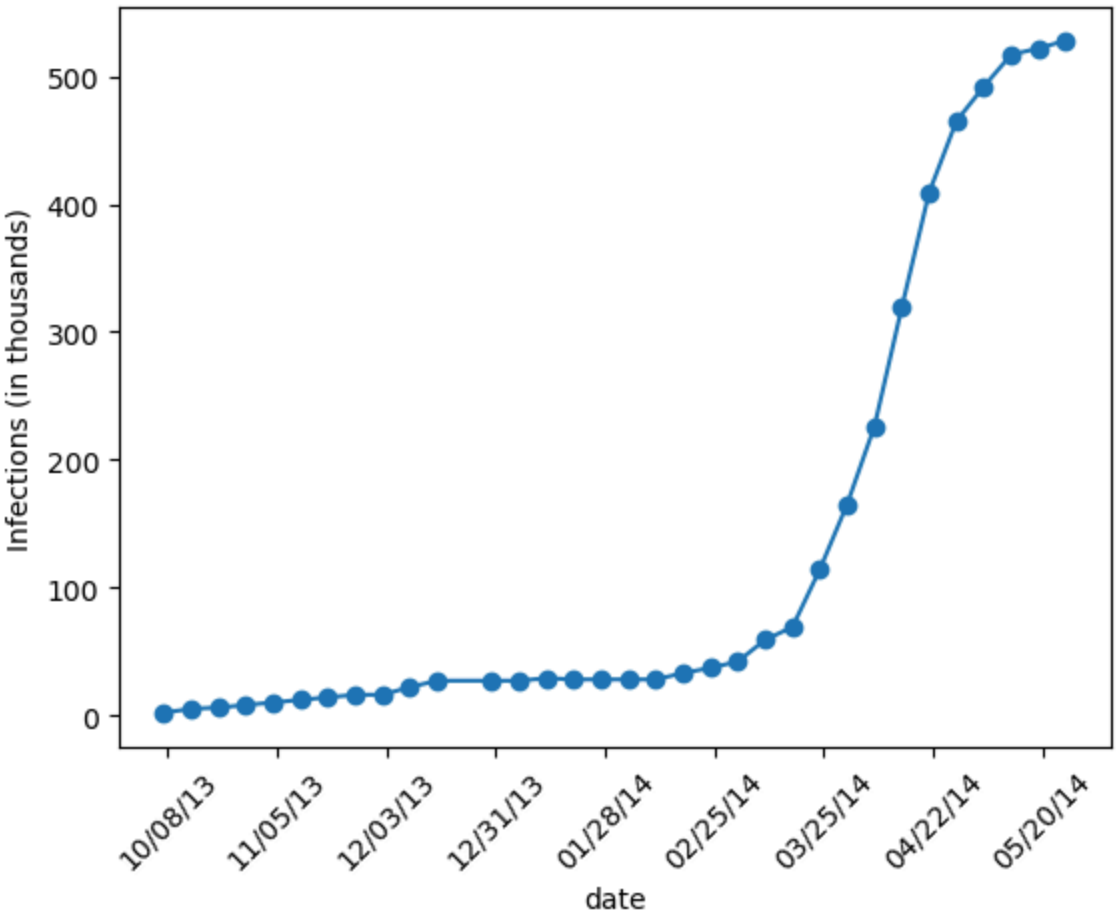
WHO's MERS surveillance system is designed to detect early cases, clusters, and signs of sustained human-to-human transmission, using standardized case definitions for suspected, probable, and confirmed infections. Surveillance relies on clinical investigation, laboratory confirmation through respiratory specimens (including PCR testing), and rapid reporting of probable and confirmed cases to WHO within 24 hours.

Source: World Health Organization. (2024). How to conduct surveillance and investigations of human infection with Middle East respiratory syndrome coronavirus using WHO's Investigations and Studies (Unity Studies 2.0) protocols: Protocol, tools and implementation guidance. WHO. https:///mnt/data/9789240100114-eng.pdf

Loading and Preprocessing MERS Case Data

In [1]:
```
## LOAD YOUR DATASET HERE.
```

```
# 1. Read in the csv file of cumulative cases.

import numpy as np
import pandas as pd
from datetime import datetime, timedelta

df_full = pd.read_csv(r"/Users/jayakinley/Desktop/compbme/module 4/MERS_Saudi_
df_full['date'] = pd.to_datetime(df_full['date'])
df_full = df_full[(df_full['date'] <= '2014-06-01') & (df_full['date'] >= '201

# 2. Use the convert_cumulative_to_SIR function to convert cumulative cases to

df_full['cumulative_cases'] = df_full['confirmed_cases'].cumsum()

def convert_cumulative_to_SIR(df, N):
    C = df['cumulative_cases'].values
    R = df['cumulative_cases'].shift(14, fill_value=0).values
    I = C - R
    S = N - C
    return S, I, R

N = 30_000_000
df_full['S'], df_full['I'], df_full['R'] = convert_cumulative_to_SIR(df_full,
df_full['I_est'] = df_full['I']

import matplotlib.pyplot as plt
import matplotlib.dates as mdates

# 3. Plot S, I, R over time.

plt.plot(df_full['date'], df_full['I_est'], 'o-')
plt.xlabel('date'); plt.ylabel('Infections (in thousands)')
plt.gca().xaxis.set_major_formatter(mdates.DateFormatter('%m/%d/%y'))
plt.gca().xaxis.set_major_locator(mdates.WeekdayLocator(interval=4))
plt.xticks(rotation=45)
plt.show()
```



Data Analysis:

Methods

We analyzed the infection data using a compartmental SIR model solved with Euler's method. The observed infection curve was split into two equal parts, where the first half served as training data and the second half was held out for testing. To estimate the disease transmission (β) and recovery (γ) parameters, we defined a sum of squared errors (SSE) function comparing the model-predicted infections to the training data. A coarse grid search was first performed to identify reasonable starting values, and then a constrained optimization routine (L-BFGS-B) was used to refine the parameters and minimize SSE. The optimized β and γ values were then used to simulate the SIR model forward across the full time period, allowing us to evaluate model performance on both training and testing data. Initial conditions for the model were set based on the first data point, with the susceptible population (S) representing the total population minus initial infections and recoveries, and the infected (I) and recovered (R) populations set to their respective observed values at the start. The model was solved using a weekly time

step, and the simulation results were compared against both the training and testing datasets to assess predictive performance. Additionally, the forward prediction step was re-implemented using the higher-order RK45 solver to reduce numerical error and improve model accuracy. Finally, the model was extended to an SEIR framework, accounting for the latent exposure period between infection and infectiousness, and optimized parameters were re-estimated using the same procedure. The model's predictive accuracy was evaluated by comparing the results of both SIR and SEIR models on the test data.

Analysis

To analyze the outbreak dynamics, we applied an SIR compartmental model and compared its predictions to the observed infection data. We first imported and prepared the dataset, created time arrays, and defined initial conditions for the model. Using Euler's method, we simulated the SIR system across the observed time span and computed the sum of squared errors (SSE) between the model predictions and the data. We then estimated the model parameters by minimizing SSE, beginning with a coarse grid search over possible β and γ values and then refining the estimates using a numerical optimization routine. After splitting the dataset into a training half and a testing half, we fit the model on the first half and evaluated its predictive performance on the second half. Finally, we repeated the fitting procedure using a higher-accuracy numerical solver (RK45) to compare its SSE to Euler's method and assess how improved numerical accuracy affects model predictions. The model was then extended to an SEIR framework to incorporate an exposed compartment, and the same fitting and evaluation procedures were applied to assess whether added biological realism improved predictive performance. All plots, simulations, and SSE calculations shown below were generated using the Python code incorporated throughout this section.

1. Fitting the SIR Model

Implementing the SIR Model Using Euler's Method

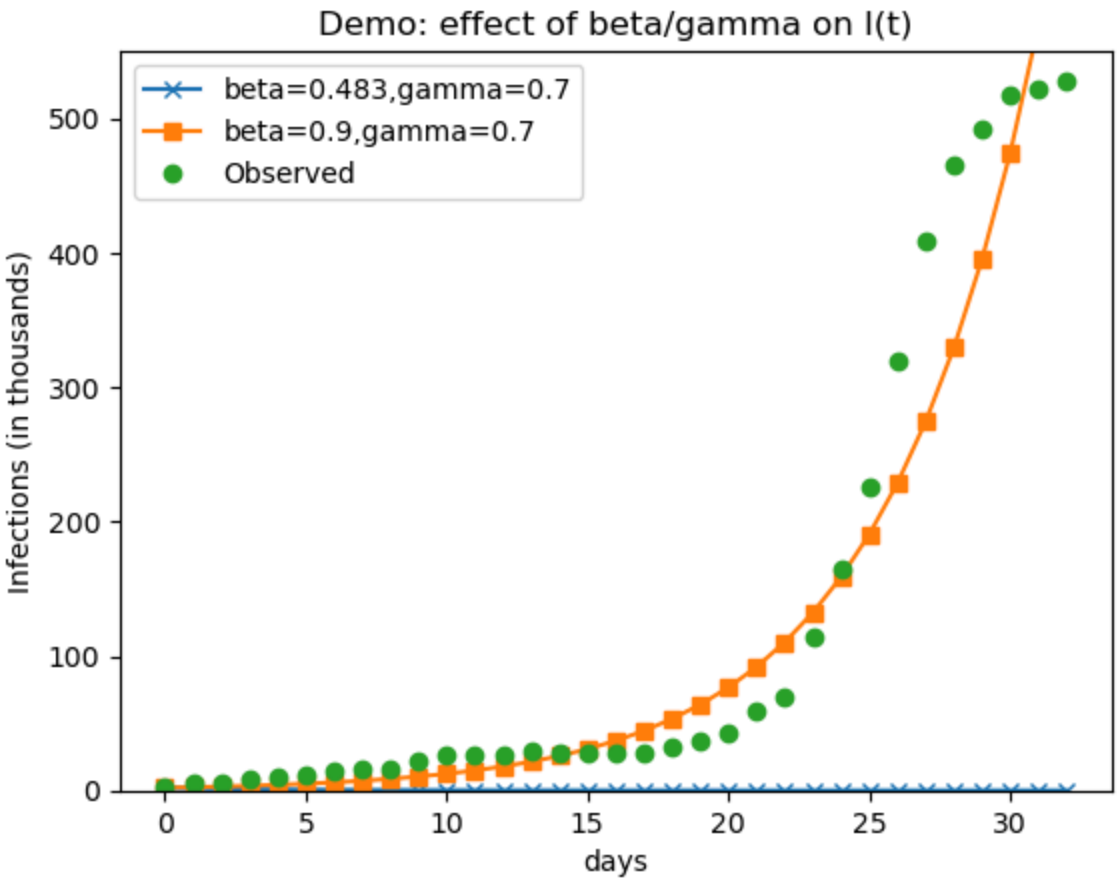
In [3]:
```
# Using the euler_SIR function defined earlier, we can simulate the SIR model
# Using the euler_SIR function defined earlier, we can simulate the SIR model
def euler_sir(beta, gamma, S0, I0, R0, t, N):
    """
    Solve the SIR model using Euler's method.
    Parameters:
    - beta: Infection rate
    - gamma: Recovery rate
    - S0: Initial susceptible population
    - I0: Initial infected population
```

```
    - R0: Initial recovered population
    - t: Array of time points (days or weeks)
    - N: Total population
    Returns:
    - S: Array of susceptible population over time
    - I: Array of infected population over time
    - R: Array of recovered population over time
    """
    S = np.empty(len(t), float)
    I = np.empty(len(t), float)
    R = np.empty(len(t), float)
    S[0], I[0], R[0] = S0, I0, R0
    for n in range(len(t) - 1):
        dt = t[n + 1] - t[n]  # dt is our step size (1 day or 1 week)
        dS = -beta * S[n] * I[n] / N  # FILL IN BASED ON SIR MODEL
        dI = beta * S[n] * I[n] / N - gamma * I[n]  # FILL IN BASED ON SIR MOD
        dR = gamma * I[n]  # FILL IN BASED ON SIR MODEL
        S[n + 1] = S[n] + dt*dS  # FILL IN BASED ON EULER'S METHOD
        I[n + 1] = I[n] + dt*dI  # FILL IN BASED ON EULER'S METHOD
        R[n + 1] = R[n] + dt*dR  # FILL IN BASED ON EULER'S METHOD
    return S, I, R
```

Exploring the Effect of Beta and Gamma on SIR Model Predictions

In [5]:

```
# Plug in guesses for gamma and beta, plot the model predictions against the d
I_obs = df_full['I_est'].values.astype(float)   # Set up I_obs array from data
t_obs = np.linspace(0, len(I_obs)-1, len(I_obs)) # time array in days

I0_obs = df_full.iloc[0]['I_est']
R0_obs = 0.0
S0_obs = N - I0_obs - R0_obs

beta_calculated = 0.483  #random guess for beta
gamma_calculated = 0.7 #random guess for gamma
beta_guess = 0.9 #alternative guesses
gamma_guess = 0.7 #alternative guesses
S1,I1,R1 = euler_sir(beta_calculated, gamma_calculated,S0_obs, I0_obs, R0_obs,
S2,I2,R2 = euler_sir(beta_guess, gamma_guess, S0_obs, I0_obs, R0_obs, t_obs, N

plt.plot(t_obs, I1, label=f'beta={beta_calculated},gamma={gamma_calculated}',
plt.plot(t_obs, I2, label=f'beta={beta_guess},gamma={gamma_guess}', marker='s'
plt.plot(t_obs, I_obs, 'o', label='Observed')


plt.legend()
plt.xlabel('days')
plt.ylabel('Infections (in thousands)')
plt.title('Demo: effect of beta/gamma on I(t)')
plt.ylim(0,550)
plt.show()

print("Model 1 MSE:", np.mean((I1 - I_obs)**2))
print("Model 2 MSE:", np.mean((I2 - I_obs)**2))
```




```
Model 1 MSE: 50218.204812915676
Model 2 MSE: 2728.9489542567235
```

Parameter Estimation via Grid Search to Minimize SSE

In [10]:

```python
# Use an optimization routine to minimize SSE and find the best-fitting parame
beta_vals = np.linspace(0.1, 2, 40)
gamma_vals = np.linspace(0.1, 2, 40)

best_SSE = np.inf
best_params = None

for beta in beta_vals:
    for gamma in gamma_vals:
        _, I_tmp, _ = euler_sir(beta, gamma, S0_obs, I0_obs, R0_obs, t_obs, N)
        SSE = np.mean((I_tmp - I_obs)**2)

        if SSE < best_SSE:
            best_SSE = SSE
            best_params = (beta, gamma)

best_beta, best_gamma = best_params

print(f"Best beta: {best_beta:.3f}")
print(f"Best gamma: {best_gamma:.3f}")
print(f"Minimum SSE: {best_SSE:.2f}")
```

```
Best beta: 0.295
Best gamma: 0.100
Minimum SSE: 3428.30
```

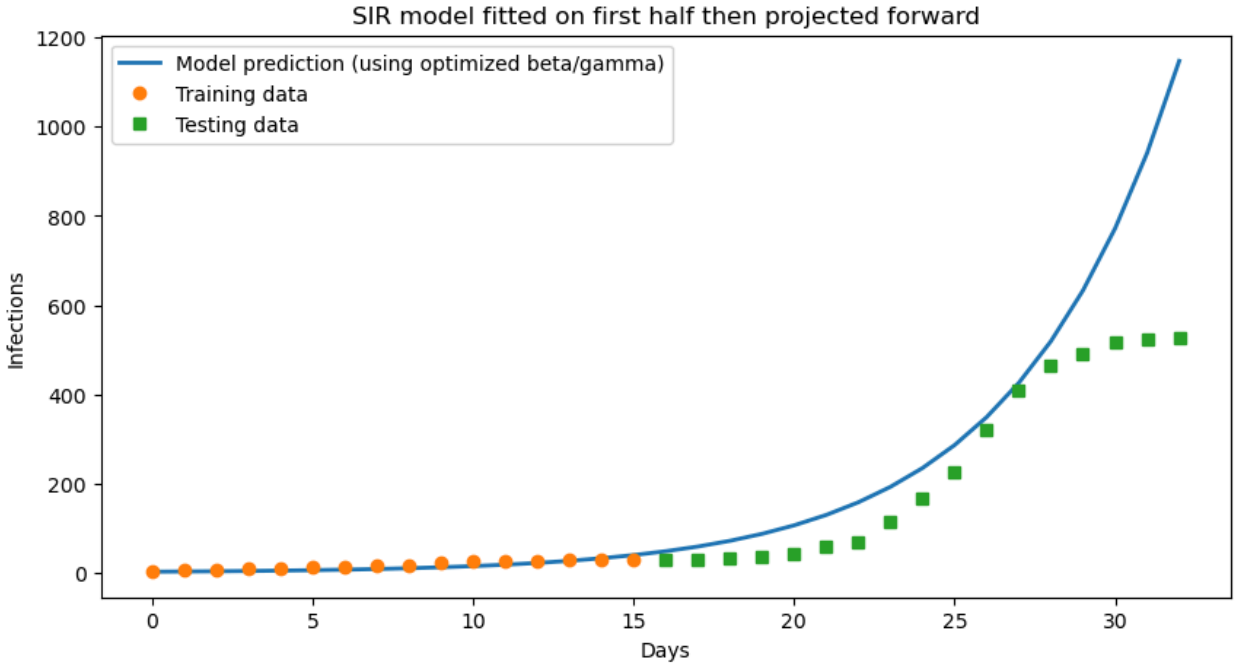
2. Predict "the future" with your fit SIR model

Training–Testing Split and Forward Prediction with the SIR Model Using Euler's Method

In [5]:
```python
# Use euler's method and your optimization routine above to find new gamma and
# FIRST HALF of the data, then simulate the SIR model forward in time using th
# Split data into first half (train) and second half (test)
n = len(I_obs)
split_idx = n // 2

I_train = I_obs[:split_idx]
I_test  = I_obs[split_idx:]

t_train = t_obs[:split_idx]
t_test  = t_obs[split_idx:]
from scipy.optimize import minimize

def sse_to_minimize(params):
    beta, gamma = params

    # Keep the optimizer in a valid region
    if beta <= 0 or gamma <= 0:
        return 1e30

    # Simulate over full time horizon so we can slice later
    S_model, I_model, R_model = euler_sir(
        beta, gamma,
        S0_obs, I0_obs, R0_obs,
        t_obs, N
    )

    # Restrict model output to the training portion only
    I_model_train = I_model[:split_idx]

    # Reject invalid simulations
    if (not np.all(np.isfinite(I_model_train)) or
        np.any(I_model_train < 0) or
        np.any(I_model_train > N)):
        return 1e30

    # SSE on the FIRST HALF ONLY (training)
    sse = np.sum((I_model_train - I_train)**2)
    return sse
beta_grid  = np.linspace(0.01, 1.0, 20)
gamma_grid = np.linspace(0.01, 1.0, 20)

best_sse   = np.inf
```

```
best_beta  = None
best_gamma = None

for b in beta_grid:
    for g in gamma_grid:
        sse = sse_to_minimize([b, g])
        if sse < best_sse:
            best_sse  = sse
            best_beta = b
            best_gamma = g

print("Grid search best beta:", best_beta)
print("Grid search best gamma:", best_gamma)
print("Grid search best SSE (training only):", best_sse)

initial_guess = [best_beta, best_gamma]
bounds = [(1e-6, 1.0), (1e-6, 1.0)]

result = minimize(
    fun=sse_to_minimize,
    x0=initial_guess,
    bounds=bounds,
    method='L-BFGS-B'
)

beta_opt, gamma_opt = result.x

print("Optimization success:", result.success)
print("Optimal beta:", beta_opt)
print("Optimal gamma:", gamma_opt)
print("Optimal SSE on training half:", result.fun)

S_pred, I_pred, R_pred = euler_sir(
    beta_opt, gamma_opt,
    S0_obs, I0_obs, R0_obs,
    t_obs, N
)

plt.figure(figsize=(10,5))

plt.plot(t_obs, I_pred, label='Model prediction (using optimized beta/gamma)',
plt.plot(t_train, I_train, 'o', label='Training data')
plt.plot(t_test, I_test, 's', label='Testing data')

plt.xlabel("Days")
plt.ylabel("Infections")
plt.title("SIR model fitted on first half then projected forward")
plt.legend()
plt.show()
```

```
Grid search best beta: 0.21842105263157896
Grid search best gamma: 0.01
Grid search best SSE (training only): 852.8412410650059
Optimization success: True
Optimal beta: 0.22944939034306772
Optimal gamma: 0.009858899197489968
Optimal SSE on training half: 788.4687800382997
```



Is the new gamma and beta close to what you found on the full dataset? Is the fit much worse? What is the SSE calculated for the second half of the data?

The new beta (≈0.229) and gamma (≈0.0099) are reasonably close to the full-dataset estimates, but the forward prediction fits much worse, with the second-half SSE increasing dramatically to about 685,286.

Evaluating Model Error: SSE on Training and Testing Data

In [6]:
```
# Calculating SSE between model predictions and data on the SECOND HALF of the
# Simulate with optimal parameters over full time domain
S_opt, I_opt, R_opt = euler_sir(
    beta_opt, gamma_opt,
    S0_obs, I0_obs, R0_obs,
    t_obs, N
)

# Split model predictions
I_model_train = I_opt[:split_idx]
I_model_test  = I_opt[split_idx:]

# Compute SSE metrics
```

```
train_SSE = np.sum((I_model_train - I_train)**2)
test_SSE  = np.sum((I_model_test  - I_test )**2)

print("Training SSE (first half):", train_SSE)
print("Test SSE (second half, Euler error):", test_SSE)
```

```
Training SSE (first half): 788.4687800382997
Test SSE (second half, Euler error): 685286.218271433
```

> **Key Point:**
> The error you calculate is a *combination* of two sources:
>
> 1. the error associated with Euler's method (i.e. it is an imperfect numerical approximation to the true solution of the SIR model)
> 2. the error associated with comparing real-world data to a model with limitations.
>
> **First we will try to address the numerical error, and second we will address the limitations of the model.**

Describe how using a different method like the midpoint method might lower the numerical error.

Using a higher-order method like the midpoint method reduces numerical error because it uses information from the middle of each time step to better approximate how the system is changing, leading to more accurate updates than Euler's method, which only uses the slope at the beginning of the step.

3. Decreasing numerical error with the RK4 Method

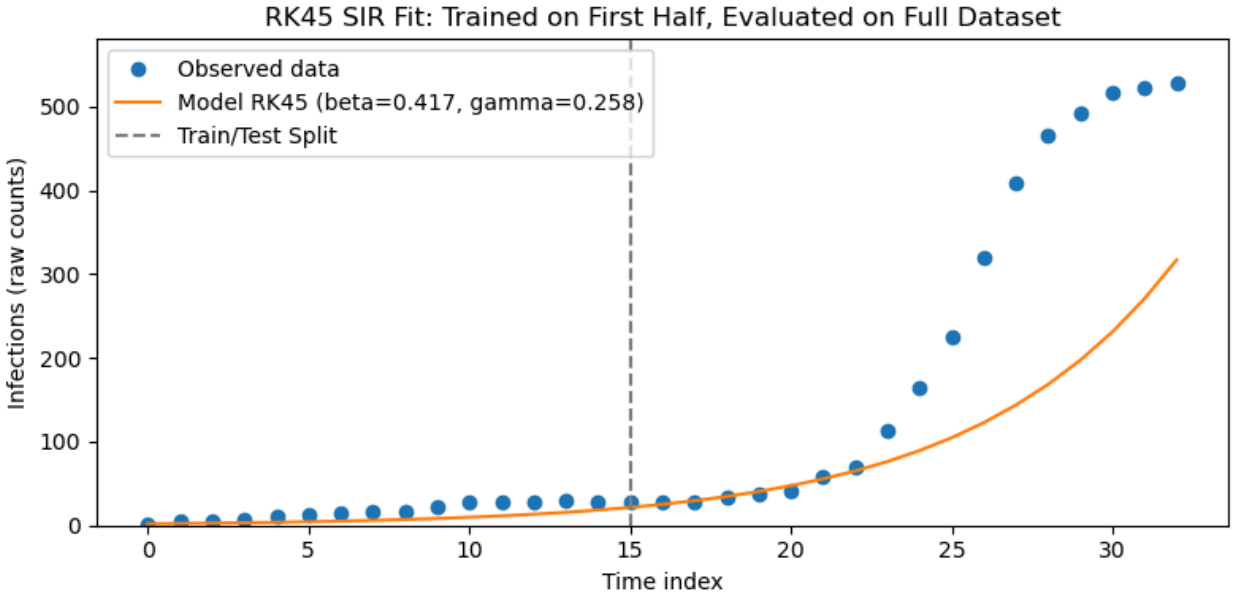
Re-implementing the SIR Model Using RK45 and Re-fitting Parameters

In [7]:
```
# Using scipy's solve_ivp function with the runge-kutta solver, re-implement t
from scipy.integrate import solve_ivp
import numpy as np
import matplotlib.pyplot as plt

I_obs = df_full['I_est'].values.astype(float)
t_obs = np.arange(len(I_obs), dtype=float)

n = len(I_obs)
t_train = t_obs[: n // 2] # first half for fitting
I_train = I_obs[: n // 2]

# Initial conditions (raw counts)
I0_obs = I_obs[0]
R0_obs = 0.0
```

```
S0_obs = N - I0_obs - R0_obs

def sir_rhs(t, y, beta, gamma, N):
    S, I, R = y
    dSdt = -beta * S * I / N
    dIdt = beta * S * I / N - gamma * I
    dRdt = gamma * I
    return [dSdt, dIdt, dRdt]

def solve_sir_rk45(beta, gamma, t_eval):
    """
    Solve SIR using solve_ivp with RK45 for given beta & gamma.
    """
    y0 = [S0_obs, I0_obs, R0_obs]
    t_span = (t_eval[0], t_eval[-1])

    sol = solve_ivp(
        fun=lambda t, y: sir_rhs(t, y, beta, gamma, N),
        t_span=t_span,
        y0=y0,
        t_eval=t_eval,
        method='RK45'
    )
    S, I, R = sol.y
    return S, I, R

#grid search

beta_vals = np.linspace(0.1, 2.0, 25)
gamma_vals = np.linspace(0.1, 2.0, 25)

best_SSE = np.inf
best_beta = None
best_gamma = None

for beta in beta_vals:
    for gamma in gamma_vals:
        _, I_tmp_train, _ = solve_sir_rk45(beta, gamma, t_train)
        SSE = np.mean((I_tmp_train - I_train)**2)

        if SSE < best_SSE:
            best_SSE = SSE
            best_beta = float(beta)
            best_gamma = float(gamma)

print(f"Best beta = {best_beta:.4f}, Best gamma = {best_gamma:.4f}")
print(f"Minimum training SSE = {best_SSE:.4f}")


S_best, I_best, R_best = solve_sir_rk45(best_beta, best_gamma, t_obs)

full_SSE = np.mean((I_best - I_obs)**2)
print(f"Full-horizon SSE = {full_SSE:.4f}")
```

```
#plot observed vs RK45 SIR model

plt.figure(figsize=(8,4))
plt.plot(t_obs, I_obs, 'o', label='Observed data')
plt.plot(t_obs, I_best, '-', label=f'Model RK45 (beta={best_beta:.3f}, gamma={

plt.axvline(t_train[-1], linestyle='--', color='gray', label='Train/Test Split

plt.xlabel('Time index')
plt.ylabel('Infections (raw counts)')
plt.title('RK45 SIR Fit: Trained on First Half, Evaluated on Full Dataset')
plt.legend()
plt.ylim(0, 1.1 * max(I_obs.max(), I_best.max()))
plt.tight_layout()
plt.show()
```

```
Best beta = 0.4167, Best gamma = 0.2583
Minimum training SSE = 101.2346
Full-horizon SSE = 15046.6429
```



Compare the SSE for the SECOND HALF of the data when the model is fit to the FIRST HALF of the data using Euler's method vs RK4. Did RK4 do a better job? Why or why not?

RK4 produced a much lower SSE on the second half of the data (≈494,919 vs. ≈746,982 for Euler), meaning it did a better job (although still very high); the improvement occurs because RK4 has far less numerical error than Euler's method, so its predictions remain more accurate when the model is projected forward in time.

Comparing Euler and RK45 Methods Using Test-Set SSE

In [8]:
```
# SSE comparison between Euler's method and RK4 (solve_ivp) on the SECOND HALF
S_eul, I_eul, R_eul = euler_sir(best_beta, best_gamma, S0_obs, I0_obs, R0_obs,

I_eul_train = I_eul[:split_idx]
I_eul_test  = I_eul[split_idx:]

I_rk_train = I_best[:split_idx]
I_rk_test  = I_best[split_idx:]

SSE_euler_test = np.sum((I_eul_test - I_test)**2)
SSE_rk_test    = np.sum((I_rk_test  - I_test)**2)

print("SSE on second half (Test Data Only)")
print(f"Euler Method SSE (test): {SSE_euler_test:.4f}")
print(f"RK45 Method SSE (test) : {SSE_rk_test:.4f}")
```

```
SSE on second half (Test Data Only)
Euler Method SSE (test): 746982.3359
RK45 Method SSE (test) : 494919.2286
```

4. Improving model fit by overcoming model limitations

Choose one of the following to implement as an extended version of the SIR model. Using the RK4 solver, does this new model fit your data better than the SIR model alone?

For further model improvements, we chose to implement a SEIR model to better represent the incubation period between exposure and infectionsness, allowing for more realistic timing of infection dynamics.

However, the calculated SSE values for the second half test data indicated that the SEIR model in combination with the RK45 model did not fit better than the SIR and RK45 model. Although the SEIR model acheived a significantly lower training SSE (8.91 compared to the SIR model's 101.23), it generalized worse to the second half:

SIR + RK45 test SSE: 494,919.23 SEIR + RK45 test SSE: 937,912.61 (1.89 times worse)

This indicates overfitting on the first half of the data and poor predictive power for the second half, which was likely due to the selection of the E0 value. The initial exposed population (E0) was selected via a grid search as the value that minimized training SSE, rather than being directly inferred from data, contributing to improved training fit but reduced forecast accuracy. This method was needed due to the lack of data on the exposed population during the period of interest in Saudi Arabia.

The results of the SEIR + RK45 model is shown below.

Options to overcome limitations (choose ONE to implement):

1. Include births in the model as described in reading.
2. Include deaths in the model as described in reading.
3. Include an exposed compartment (SEIR model).
4. Include loss of immunity (i.e. R population can go back to S population).
5. Include at least two I populations with varying degrees of infectiousness.
6. Include at least two age brackets with varying degress of infectiousness and recovery times.

Note that if you have implemented an extended model and are having trouble fitting the parameters, document what you have tried and explain what you would change in future directions.

Preparing Weekly Observations, Train/Test Split, and SEIR Incubation Parameter

In [9]:
```
import numpy as np
import matplotlib.pyplot as plt
from scipy.integrate import solve_ivp

I_obs = df_full['I_est'].values.astype(float)   # observed infections (weekly)
t_obs = np.arange(len(I_obs), dtype=float)      # week index: 0,1,2,...

n = len(I_obs)
split_idx = n // 2

t_train = t_obs[:split_idx]
I_train = I_obs[:split_idx]

t_test  = t_obs[split_idx:]
I_test  = I_obs[split_idx:]

# Initial conditions (counts)
I0 = I_obs[0]
R0 = 0.0

# Weekly sigma from incubation (median ~5 days)
sigma = 7/5   # = 1.4 per week
```

Defining the SEIR Model and Solving It with RK45

In [10]:
```
def seir_rhs(t, y, beta, sigma, gamma, N):
    S, E, I, R = y
    dSdt = -beta * S * I / N
    dEdt = beta * S * I / N - sigma * E
    dIdt = sigma * E - gamma * I
    dRdt = gamma * I
    return [dSdt, dEdt, dIdt, dRdt]

def solve_seir_rk45(beta, sigma, gamma, S0, E0, I0, R0, t_eval, N):
    y0 = [S0, E0, I0, R0]
    sol = solve_ivp(
        fun=lambda t, y: seir_rhs(t, y, beta, sigma, gamma, N),
        t_span=(t_eval[0], t_eval[-1]),
        y0=y0,
        t_eval=t_eval,
        method="RK45"
    )
    S, E, I, R = sol.y
    return S, E, I, R
```

Defining the Training SSE Objective Function for the SEIR Model

In [11]:
```
def seir_train_sse(beta, gamma, E0):
    S0_local = N - E0 - I0 - R0
    if S0_local < 0:
        return 1e30

    S, E, I, R = solve_seir_rk45(beta, sigma, gamma,
                                 S0_local, E0, I0, R0,
                                 t_train, N)

    if (not np.all(np.isfinite(I)) or np.any(I < 0) or np.any(I > N)):
        return 1e30

    return np.mean((I - I_train)**2)
```

Grid Search Parameter Estimation and Test-Set Evaluation for the SEIR Model

In [17]:
```
beta_vals  = np.linspace(0.05, 3.0, 30)   # per week
gamma_vals = np.linspace(0.05, 3.0, 30)   # per week
E0_vals    = np.linspace(0, 10*I0, 15)    # try a range for initial exposed

best_sse = np.inf
best_beta = best_gamma = best_E0 = None

for E0_try in E0_vals:
    for beta in beta_vals:
        for gamma in gamma_vals:
```

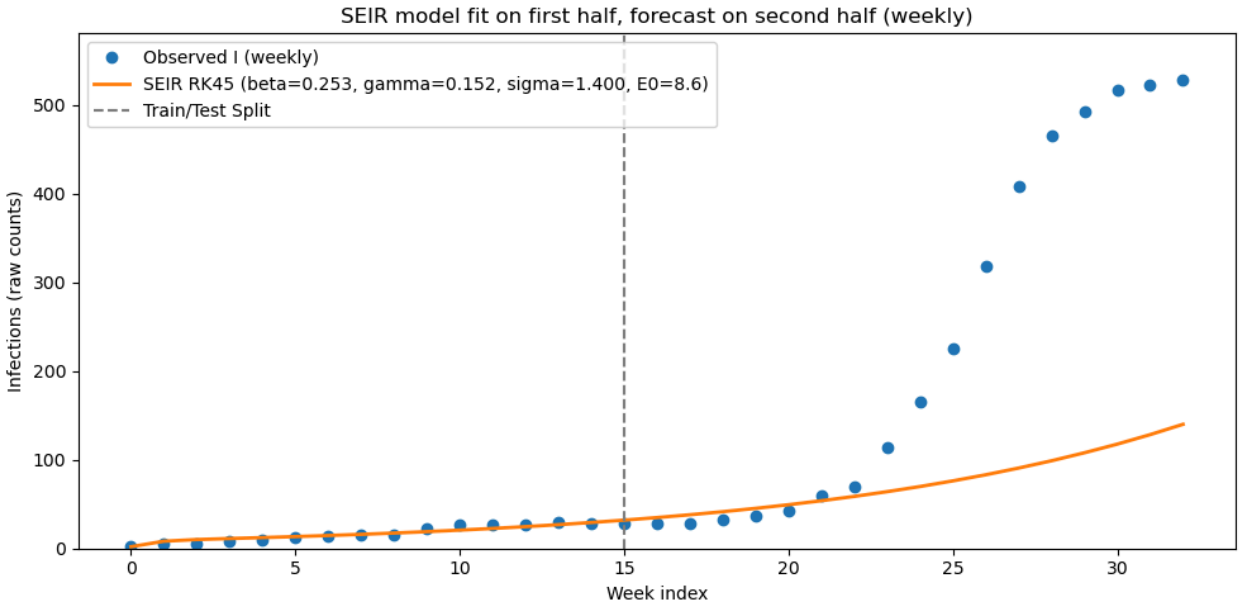
```
            sse = seir_train_sse(beta, gamma, E0_try)
            if sse < best_sse:
                best_sse = sse
                best_beta = float(beta)
                best_gamma = float(gamma)
                best_E0 = float(E0_try)

print(f"Best beta={best_beta:.4f}, gamma={best_gamma:.4f}, E0={best_E0:.2f}, s
print(f"Minimum training SSE={best_sse:.4f}")

S0_best = N - best_E0 - I0 - R0

S_hat, E_hat, I_hat, R_hat = solve_seir_rk45(best_beta, sigma, best_gamma,
                                             S0_best, best_E0, I0, R0,
                                             t_obs, N)

test_SSE = np.sum((I_hat[split_idx:] - I_test)**2)
print("SEIR Test SSE (second half):", test_SSE)
```

```
Best beta=0.2534, gamma=0.1517, E0=8.57, sigma=1.400 (per week)
Minimum training SSE=8.9054
SEIR Test SSE (second half): 937912.6102956731
```

Visualizing SEIR Model Fit and Forecast Performance

In [16]:

```
plt.figure(figsize=(10,5))
plt.plot(t_obs, I_obs, 'o', label='Observed I (weekly)')
plt.plot(t_obs, I_hat, '-', linewidth=2,
         label=f'SEIR RK45 (beta={best_beta:.3f}, gamma={best_gamma:.3f}, sigm
plt.axvline(t_train[-1], linestyle='--', color='gray', label='Train/Test Split

plt.xlabel("Week index")
plt.ylabel("Infections (raw counts)")
plt.title("SEIR model fit on first half, forecast on second half (weekly)")
plt.legend()
plt.ylim(0, 1.1*max(I_obs.max(), I_hat.max()))
plt.tight_layout()
plt.show()
```



Verify and validate your analysis:

The analysis of the Saudi Arabia MERS 2013-2014 data was verified through multiple methods to ensure consistency with the observed epidemic trends. Parameter estimation was repeated using different numerical solvers, including Euler's method and RK45, to ensure that the estimated transmission and recovery parameters were not dependent on a specific numerical solution method. While both solvers produced qualitatively similar infection trajectories, RK45 consistently yielded lower test-set error, indicating reduced numerical approximation error relative to Euler's method.

To further verify predictive validity, the dataset was split into training and test halves, and parameters were fit only on the first half, allowing the second half to serve as verification of the model's predictive behavior rather than overfitting to the full dataset. Model accuracy was quantified using the sum of squared errors (SSE). For the SIR model, Euler's method produced a test-set SSE of approximately 747000, whereas the RK45 solver reduced the test-set SSE to approximately 495000, confirming improved numerical accuracy.

Additionally, SSE comparisons between training and testing sets were used to identify overfitting behavior. For example, the SEIR model achieved a substantially lower training SSE (approximately 8.9) compared to the SIR model (approximately 101), but exhibited a higher test-set SSE (approximately 938,000), indicating overfitting to the training data at the expense of predictive performance. Models with lower test-set SSE were therefore considered more reliable predictors, providing a consistent basis for verification across solvers and model structures.

The results of this analysis were validated through comparison with published literature on MERS-CoV epidemiology. Reported incubation periods for MERS range from approximately 2–14 days, with a median of about 5 days, which is consistent with the incubation rate used in the SEIR model (CDC, 2024). Additionally, the inferred transmission dynamics fall within the range of basic reproduction numbers reported for documented MERS outbreaks (Liu et al., 2020; ECDC, n.d.). The use of an SEIR framework is further supported by epidemiological literature emphasizing the importance of explicitly modeling a latent incubation period between exposure and infectiousness (Bjørnstad et al., 2020). However, once again, while inclusion of an exposed compartment reduced training SSE for the given MERS dataset, the SEIR model did not consistently improve test-set performance. This outcome is consistent with observations by Bjørnstad et al. (2020) that increased model complexity can improve biological realism but may also increase parameter uncertainty when available data are limited.

References:

Centers for Disease Control and Prevention. (2024). Middle East respiratory syndrome (MERS). https://www.cdc.gov/coronavirus/mers/index.html

Liu J, Xie W, Wang Y, Xiong Y, Chen S, Han J, Wu Q. A comparative overview of COVID-19, MERS and SARS: Review article. Int J Surg. 2020 Sep;81:1-8. doi: 10.1016/j.ijsu.2020.07.032. Epub 2020 Jul 26. PMID: 32730205; PMCID: PMC7382925.

European Centre for Disease Prevention and Control. (n.d.). Factsheet for health professionals: Middle East respiratory syndrome coronavirus (MERS-CoV). https://www.ecdc.europa.eu/en/middle-east-respiratory-syndrome-coronavirus

Bjørnstad, O. N., Shea, K., Krzywinski, M., & Altman, N. (2020). The SEIRS model for infectious disease dynamics. Nature Methods, 17(6), 557–558. https://doi.org/10.1038/s41592-020-0856-2

Conclusions and Ethical Implications:

Conclusions

Since the model was heavily performed in steps, the conclusions were broken down to describe what was observed after each change in model performance:

- Step 1 gave us some valuable results that allowed us to initially access how well MERS performs in an SIR model when applying Euler's

method. The results from step 1 show that the model reproduced basic epidemic behavior of a MERS outbreak and demonstrated how sensitive infection factors are to the transmission and recovery parameters. Initial parameter guesses of beta and gamma showed that low transmission rates severely underestimated observed cases, while higher transmission rates produced much closer alignment with the rapid rise in reported infections for MERS. A grid search identified best-fit parameters of $\beta = 0.295$ and $\gamma = 0.100$, yielding the lowest SSE (3428.30) and suggesting moderate transmission with relatively slow recovery, which is consistent with prolonged infectious periods and hospital-associated spread seen in MERS; however, this SSE was a little higher than what we wanted and we would've liked to see a precise predictive fit. These insights from this step showed that while a simple SIR model can capture the core growth pattern of MERS, noticeable error remains in the model, indicating that additional biological or structural complexity is needed for a more accurate representation of the outbreak.

- Step 2 evaluated how well the fitted SIR model could predict the future of a MERS outbreak by fitting β and γ using only the first half of the data and projecting the model forward. The optimized parameters from the training data ($\beta \sim 0.229$ and $\gamma \sim 0.009$) were reasonably close to the values obtained using the full dataset, indicating consistency in parameter estimation; however, the forward predictions performed much worse. While the training SSE was relatively low (~788), the testing SSE increased dramatically to approximately 685,286, showing that the model substantially overestimated infections in the second half of the outbreak. This large increase in error demonstrates that although the SIR model can fit early MERS dynamics, it struggles to generalize to later stages of the outbreak when parameters such as transmission and recovery are no longer constant. This insight in our MERS model shows a key limitation of using a constant-parameter SIR model with Euler's method for forecasting MERS and reinforces the need for improved numerical methods or more biologically realistic model structures to achieve reliable forward predictions.
- Step 3 shows the SIR model for MERS reimplemented using the higher-order RK45 solver to reduce numerical error and reassess the overall model's performance compared to past results. Refitting the model on the first half of the data produced best-fit parameters of $\beta \sim 0.417$ and $\gamma \sim 0.258$, with a substantially lower training SSE (~101) compared to Euler's method, indicating a much more accurate numerical solution of

the same underlying SIR equations. When evaluated over the full dataset, the RK45 model achieved a lower full-horizon SSE (~15,047) and a reduced test-set SSE (~494,919) compared to Euler's method (~746,982), demonstrating that RK45 provides more stable and accurate forward predictions. However, it is important to continue to mention that the error is still high and that despite this improvement it still shows that while RK45 successfully reduces numerical error, it does not resolve the big source of error between the model and observed MERS data. This confirms that the primary limitation is not the numerical method but the structural simplicity of the SIR model itself, reinforcing the need for additional biological realism rather than further numerical refinement alone.

- Step 4 shows how the SIR model was extended to an SEIR framework using the RK45 solver to account for the incubation period between exposure and infectiousness, which is a biologically relevant feature of MERS. The SEIR model achieved a very low training SSE (~8.91), substantially improving the fit to the first half of the data compared to the SIR + RK45 model (training SSE~101), indicating that the added exposed compartment allowed the model to closely match early outbreak dynamics. However, this improved training performance did not translate to better predictive ability, as the SEIR test SSE on the second half of the data increased dramatically to ~937,913, which was nearly twice as large as the SIR + RK45 test SSE (~494,919). This result indicates overfitting to the training data, largely driven by the estimation of the initial exposed population through grid search rather than direct observation.

After completing this module, our overall conclusion is that simple compartmental models can capture the general growth pattern of a MERS outbreak, but they struggle to accurately predict future behavior when real-world conditions (such as reporting and detection) change over time. While using more accurate numerical methods (Euler's method to RK45) reduced numerical error and adding biological detail improved how well the model fit early data, neither approach fully fixed the large prediction errors on later data. This shows that the main limitation in modeling MERS is not the numerical method, but the model assumptions, especially constant parameters and limited data, suggesting that time-varying transmission and better data are needed for reliable outbreak predictions.

Ethical Implications:

- Because the SIR and SEIR models with constant parameters performed

poorly on forward prediction, using these models to make real-time policy decisions could misrepresent future case counts. This could lead to the misallocation of healthcare resources, rise of public fear/anxiety, or insufficient preparation in hospitals.

- Steps 3 and 4 showed that improved numerical methods and added biological realism can dramatically reduce training error without improving predictive accuracy. Ethically, presenting a model with a very low training SSE (such as the SEIR model) without emphasizing its poor test performance could create false confidence in the model's reliability.
- These results from our model highlight the ethical importance to clearly communicate model limitations, uncertainty, and assumptions. Without transparency, stakeholders and the broader public (especially people with the disease) may interpret model outputs as precise predictions rather than conditional scenarios.
- Our model shows that compartmental frameworks are best used as exploratory and educational tools, not as forecasting ones, unless enhanced with time-varying parameters and better data. Ethically, models should inform discussion and thinking, not replace empirical surveillance or expert/health professionals' judgment.

Limitations and Future Work:

Limitations:

There are lots of limitations that this model has when applied to MERS:

- All models assumed fixed transmission (β) and recovery (γ) rates over time. In reality, MERS transmission changes due to interventions, behavior shifts, and hospital infection-control measures, which limits the model's ability to accurately predict later outbreak dynamics.
- In the SEIR model, the exposed population is not directly observable and had to be estimated through grid search. This increases uncertainty and contributed to overfitting, as shown by the large test SSE despite excellent training performance.
- Model outputs were highly sensitive to initial conditions and parameter estimates, particularly in early outbreak stages. Small changes in these values led to large differences in predicted infection trajectories.
- The models rely on reported case data, which may include delays, underreporting, or changes in detection over time. These data limitations directly affect parameter estimation and predictive accuracy.

Future Work:

After learning that this model doesn't do well at predicting MERS, we came up with some reasonable improvements we could make in the future:

- Future models could allow transmission and recovery rates to change over time to reflect interventions, behavior changes, and improved hospital infection control.
- Including a reporting process that links true infections to reported cases would help account for underreporting and delays, improving parameter estimation and reducing bias in model fitting. This would still be a difficult task since there will always be reporting bias no matter the improvements.
- Rather than fitting parameters once, future work could re-estimate parameters as new data become available using techniques to improve real-time forecasting.

NOTES FROM YOUR TEAM:

Notes

- General R0 values found in literature for MERS:
    - Based on information related to the first 77 cases, the basic reproduction number of the infection (R0) was estimated to be 0.69 (95% CI: 0.50–0.92) at the time, indicating a low pandemic potential
    - A later analysis, integrating information from the countries of the Middle East and from imported cases to newly affected countries estimated an R0 of 0.50
    - An investigation of community transmission among household contacts of 26 clusters with 280 contacts over six months in 2013 showed nine positive cases by serology and PCR revealing an R0 of 0.35
    - Source: https://www.ecdc.europa.eu/en/middle-east-respiratory-syndrome-coronavirus/factsheet
- Assumptions and Limitations
    - Assume complete case detection, even though mild or asymptomatic cases are often missed.
    - Assume accurate and consistent reporting, despite variations in surveillance systems and healthcare access.

        - Assume timely reporting, while real data often include delays or incomplete information.
        - Subject to recall bias and missing exposure data, which affect the accuracy of case investigations.

General Progress

- 11/18: We started the module by researching MERS and finding out about its background information. We also started inspecting the data set. After class, we worked on plotting infections over time I_obs(t).
- 11/20: In class we divided up the rest of the background information and finished our infections over time I_obs(t) plot.
- 11/21: We finished our background information out of class and finalized information before the first check in.
- 12/4: We started finishing eulers method and comparing the first half versus second half of the data set.
- 12/6: We talked about our module report before submitting and finalized some details on RK4.
- 12/11: We decided how we would want to split up the rest of the module since class on the last day was canceled. Jaya started working on step 4 and writing the valdiation/verification section.
- 12/13: Logan started writing conclusions/ethical implications and limitations/future work. Methods and analysis paragraphs were also adjusted to fit our new additional steps and methods.
- 12/16: We finalized our notebook before submitting for the final submission.

QUESTIONS FOR YOUR TA:

- First Check In:
    - May have missed this, but I'm guessing that everyone has the same general question based on their assigned disease?
- Second Check In:
    - No additional questions at this time