Inner Product

- If **u** and **v** are vectors in $\mathbb{R}^n$, then we regard **u** and **v** as $n \times 1$ matrices.

$$\mathrm{u} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} \qquad \mathrm{v} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix}$$

- The inner product (dot product) of **u** and **v** is

$$: u_1 v_1 + u_2 v_2 + \cdots + u_n v_n.$$

Dot Product - Example

- Example: $u = \begin{bmatrix} 1 \\ -3 \\ 4 \end{bmatrix}$ $v = \begin{bmatrix} 2 \\ 1 \\ -2 \end{bmatrix}$
- u.v = (1)(2)+(-3)(1)+(4)(-2) = 2-3-8=-9
- Find the dot product of u and v in Python.

```
import numpy as np
```

```
u = np.array([1,-3,4])
```

```
v = np.array([2,1,-2])
np.dot(u,v)
```

```
-9
```

Inner product using matrix multiplication

- Can you express the dot product in terms of matrix multiplication of u and v?
- Recall that for two matrices A and B, the number of columns in A has to be the same as the number of rows in B, for AB product to be feasible:

$$A_{m,\mathbf{n}}\, B_{\mathbf{n},p} \Rightarrow C_{m,p}$$

- Hint: The transpose $\mathbf{u}^T$ is a 1xn matrix, and the matrix product $\mathbf{u}^T\mathbf{v}$ is a 1x1 matrix, which we write as a single real number (a scalar) without brackets.

- If $\mathrm{u}=\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix}$ and $\mathrm{v}=\begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix}$, then the inner product of **u** and **v** is

$$\begin{bmatrix} u_1 & u_2 & \cdots & u_n \end{bmatrix}\begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} = u_1 v_1 + u_2 v_2 + \cdots + u_n v_n$$