A. Basics for Group and Representation Theory

This section gives the basic definitions about groups and representations necessary to understand this work. We refer to the literature for a more detailed introduction (Artin, 2011; Bröcker & Dieck, 2003).

A.1. Groups

A *group* $(G, \cdot)$ is a set G together with a function $\cdot : G \times G \to G, (g, h) \mapsto g \cdot h$ called group *operation* satisfying

1. (Associativity): $g \cdot (h \cdot i) = (g \cdot h) \cdot i$ for all $g, h, i \in G$
2. (Existence of a neutral element): There is a $e \in G$ such that: $e \cdot g = g \cdot e = g$ for all $g \in G$
3. (Existence of an inverse): For all $g \in G$, there is a g^{-1} such that $e = g^{-1} \cdot g = g \cdot g^{-1}$

If in addition, G satisfies

4. (Commutativity): $g \cdot h = h \cdot g$ for all $g, h \in G$

G is called *Abelian*. We simply write $g_1 g_2$ for $g_1 \cdot g_2$ if it is clear from the context.

If $\rho : G \to G'$ is a map between two groups, it is called a *group homomorphism* if $\rho(g \cdot g') = \rho(g) \cdot \rho(g')$. That is, the map preserves the action of the group. A *group isomorphism* is a homomorphism that is bijective. In the later case, G and G' are called isomorphic and we write $G \cong G'$.

The Euclidean group

In the context of this work, the most important example of a group is the *Euclidean group* $E(n)$ consisting of all *isometries*, i.e. the set of all functions $T : \mathbb{R}^n \to \mathbb{R}^n$ such that

$$\|T(\mathbf{x}) - T(\mathbf{x}')\| = \|\mathbf{x} - \mathbf{x}'\|, \quad \text{for all } \mathbf{x}, \mathbf{x}' \in \mathbb{R}^n$$

Defining the group operation as the composition of two isometries by $T_1 \cdot T_2 := T_1 \circ T_2$, we can identify $E(n)$ as a group.

Subgroups

A *subgroup* H of a group $(G, \cdot)$ is a subset $H \subset G$ which is closed under the action of the original group. I.e. a set $H \subset G$ is a subgroup of $(G, \cdot)$ if $h_1 \cdot h_2 \in H$ for all $h_1, h_1 \in H$ and $h^{-1} \in H$ for all $h \in H$. A subgroup is typically denoted by $H < G$.

We can identify all intuitive geometric transformations on $\mathbb{R}^n$ as subgroups of $E(n)$:

1. **Translation:** For any vector $\mathbf{x} \in \mathbb{R}^n$, a translation by $\mathbf{x}$ is given by the map $t_{\mathbf{x}} : \mathbb{R}^n \to \mathbb{R}^n, \mathbf{x}' \mapsto \mathbf{x} + \mathbf{x}'$. The group of all translations is denoted by $T(n)$.
2. **Rotoreflection:** The orthogonal group $O(n) = \{Q \in \mathbb{R}^{n \times n} | QQ^T = I\}$ describes all reflections and subsequent rotations.
3. **Rotation:** The special orthogonal group $SO(n) = \{R \in O(n) | \det R = 1\}$ describes all rotations in $\mathbb{R}^n$.

Normal subgroups

A *normal subgroup* of a group is a subgroup which is closed under conjugation of the group. That is, N is a normal subgroup of G if it is a subgroup of G and

$$gng^{-1} \in N \quad \text{for all } \ n \in N, g \in G$$

Typically a normal subgroup is denoted $N \triangleleft G$. The most important example for this work is $T(n) \triangleleft E(n)$.

Semidirect product groups

A group G is a *semidirect product* of a subgroup $H < G$ and a normal subgroup $N \triangleleft G$ if it holds that for all $g \in G$, there are unique $n \in N, h \in H$ such that $g = nh$. There are a number of equivalent conditions, but not needed for this exposition. The semidirect product of two groups is denoted by

$$G = N \rtimes H$$

Most importantly, we can identify $E(n) = T(n) \rtimes O(n)$ as the semidirect product of $T(n)$ and $O(n)$.

A.2. Representations of Groups

Group representations are a powerful tool to describe the algebraic properties of geometric transformations: Let V be a vector space and $\mathrm{GL}(V)$ be the **general linear group**, i.e. the group of all linear, invertible transformations on V with the composition $f \cdot g = f \circ g$ as group operation. Then a **representation** of a group H is a group homomorphism $\rho : H \to \mathrm{GL}(V)$. For $V = \mathbb{R}^d$, this is the same as saying a group representation is a map $\rho : H \to \mathbb{R}^{d\times d}$ such that

$$\rho(h_1 \cdot h_2) = \rho(h_1)\rho(h_2)$$

where the right hand side is typical matrix multiplication.

The simplest group representation is the *trivial representation* ρ_{triv} which maps all elements of the group to the identity,

$$\rho_{triv}(h) = \mathbf{1}_d \quad \text{for all } h \in H \tag{19}$$

Orthogonal and unitary groups

An *orthogonal representation* is a representation $\rho : H \to \mathrm{GL}(\mathbb{R}^d)$ such that $\rho(h) \in O(d)$ for all $h \in H$. For *compact groups* H, every representation is equivalent to an orthogonal representation (Bröcker & Dieck, 2003, Theorem II.1.7). This is useful as the identity $\rho(h)^T = \rho(h)^{-1}$ often makes calculations significantly easier. Since in this work we focus on subgroups $H \subset O(d)$ which are all compact, it is not a restriction to assume that.

Direct sums

Given two representations, $\rho_1 : H \to \mathrm{GL}(\mathbb{R}^n)$ and $\rho_2 : H \to \mathrm{GL}(\mathbb{R}^m)$, we can combine them together to give their *direct sum*, $\rho_1 \oplus \rho_2 : H \to \mathrm{GL}(\mathbb{R}^{n+m})$, defined by

$$(\rho_1 \oplus \rho_2)(h) = \begin{bmatrix} \rho_1(h) & 0 \\ 0 & \rho_2(h) \end{bmatrix} \tag{20}$$

i.e the block diagonal matrix comprised of the two representations. This sum generalises to an arbitrary number of representations.

Tensor products

Let V_1, V_2 be two vector spaces and $V_1 \otimes V_2$ their tensor product. Given two representations, $\rho_1 : H \to \mathrm{GL}(V_1)$ and $\rho_2 : H \to \mathrm{GL}(V_2)$, we can take the tensor product representation $\rho_1 \otimes \rho_2 : H \to \mathrm{GL}(V_1 \otimes V_2)$ defined by the condition that

$$[\rho_1 \otimes \rho_2](h)(v_1 \otimes v_2) = (\rho_1(h)v_1) \otimes (\rho_2(h)v_2) \tag{21}$$

for all $v_1 \in V_1, v_2 \in V_2, h \in H$.

To make this concrete for proposition 2, we have $V_1, V_2 = \mathbb{R}^d$ and the tensor product becomes $V_1 \otimes V_2 = \mathbb{R}^{d\times d}$ with $v_1 \otimes v_2 = v_1 v_2^T$ the outer product for all $v_1, v_2 \in \mathbb{R}^d$. Setting $\rho = \rho_1 = \rho_2$, the tensor product representation $\rho \otimes \rho$ becomes

$$[\rho \otimes \rho](h)(v_1 \otimes v_2) = (\rho(h)v_1)(\rho(h)v_2)^T = \rho(h)v_1 v_2^T \rho(h)^T \tag{22}$$

Therefore, ρ_Σ as defined in eq. (15) is the tensor product $\rho \otimes \rho$. We can return such representations to the more usual matrix-acting-on-vector format by vectorising these expressions. Using the identity $\mathrm{vec}(ABC) = [C^T \otimes_{kron} A]\,\mathrm{vec}(B)$, with $\otimes_{kron}$ being the usual Kronecker product and $\mathrm{vec}(A)$ being the column-wise vectorisation of A we get

$$\mathrm{vec}(\rho(h)A\rho(h)^T) = [\rho(h) \otimes_{kron} \rho(h)]\,\mathrm{vec}(A) \tag{23}$$

B. Proofs

B.1. Proof of proposition 1

Proposition 1. *Let P be a stochastic process over $\mathcal{F}_\rho$. Then P is G-invariant if and only if the posterior map $Z \mapsto P_Z$ is G-equivariant, i.e.*

$$P_{g.Z} = g.P_Z \quad \textit{for all } g \in G \tag{7}$$

Proof. Let us be given a distribution P over functions $\mathcal{F}_\rho$ and $F \sim P$. Define $g.P$ to be the distribution of $g.F$. For any $\mathbf{x}_1, \dots, \mathbf{x}_k \in \mathbb{R}^n$ let $\mathbf{x}_{1:k}$ denote the concatenation $(\mathbf{x}_1, \dots, \mathbf{x}_k)$ of these vectors and let $g\mathbf{x}_{1:k}$ be $(g\mathbf{x}_1, \dots, g\mathbf{x}_k)$. For any such $\mathbf{x}_{1:k}$, let $\psi^P_{\mathbf{x}_{1:k}}$ be the finite-dimensional marginal of P, i.e. the distribution such that

$$[F(\mathbf{x}_1), \dots, F(\mathbf{x}_k)]^T \sim \psi^P_{\mathbf{x}_{1:k}}$$

For simplicity, we assume here that $\psi^P_{\mathbf{x}_{1:k}}$ is absolutely continuous with respect to the Lebesgue measure, i.e. has a density $\lambda^P_{\mathbf{x}_{1:k}}$. Our proof uses Kolmogorov's theorem (Øksendal, 2000), which says that two stochastic processes coincide if and only if their finite-dimensional marginals agree. Before the actual proof, we need the following four auxiliary statements.

1. Marginals of posterior. Let $F \sim P$ and let us given a context set $Z = \{(\mathbf{x}'_i, \mathbf{y}'_i)\}_{i=1}^l$ where $\mathbf{y}'_i = F(\mathbf{x}'_i)$ for all $i = 1, \dots, l$. The posterior P_Z is again a stochastic process with marginals $\psi^{P_Z}_{\mathbf{x}_{1:k}}$ and conditional density given by

$$\lambda^{P_Z}_{\mathbf{x}_{1:k}}(\mathbf{y}_{1:k}) = \lambda^P_{\mathbf{x}_{1:k}|\mathbf{x}'_{1:l}}(\mathbf{y}_{1:k}|\mathbf{y}'_{1:l}) = \frac{\lambda^P_{\mathbf{x}_{1:k},\mathbf{x}'_{1:l}}(\mathbf{y}_{1:k}, \mathbf{y}'_{1:l})}{\lambda^P_{\mathbf{x}'_{1:l}}(\mathbf{y}'_{1:l})} \tag{24}$$

2. Marginals of transformed process. If $F \sim P$, it holds that $g.P$ has marginals $\psi^{g.P}_{\mathbf{x}_{1:k}}$ with density given by

$$\lambda^{g.P}_{\mathbf{x}_{1:k}}(\mathbf{y}_{1:k}) = \lambda^P_{g^{-1}\mathbf{x}_{1:k}}(\rho(h)^{-1}\mathbf{y}_{1:k}) \tag{25}$$

after using a change of variables.

3. Express invariance in terms of marginals. By definition, P is G-invariant if $g.P = P$ for all $g \in G$. By Kolmogorov's theorem, this is equivalent to the fact the finite-dimensional marginals of P and $g.P$ agree for all $g \in G$, i.e.

$$\psi^P_{\mathbf{x}_{1:k}} = \psi^{g.P}_{\mathbf{x}_{1:k}} \quad \text{for all } \mathbf{x}_1, \dots, \mathbf{x}_k \in \mathbb{R}^n \tag{26}$$

$$\Leftrightarrow \lambda^P_{\mathbf{x}_{1:n}}(\mathbf{y}_{1:n}) = \lambda^{g.P}_{\mathbf{x}_{1:n}}(\mathbf{y}_{1:n}) = \lambda_{g^{-1}\mathbf{x}_{1:n}}(\rho(h)^{-1}\mathbf{y}_{1:n}) \quad \text{for all } \mathbf{y}_1, \dots, \mathbf{y}_k \in \mathbb{R}^d, \mathbf{x}_1, \dots, \mathbf{x}_k \in \mathbb{R}^n \tag{27}$$

where we used eq. (25) in the last equation.

4. Express equivariance in terms of marginals. Next, let us be given a context set $Z = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$ where $\mathbf{y}_i = F(\mathbf{x}_i)$. We compute:

$$P_{g.Z} = g.P_Z \tag{28}$$

$$\Leftrightarrow \psi^{P_{g.Z}}_{\mathbf{x}_{1:k}} = \psi^{g.P_Z}_{\mathbf{x}_{1:k}} \quad \text{for all } \mathbf{x}_1, \dots, \mathbf{x}_k \in \mathbb{R}^n \tag{29}$$

$$\Leftrightarrow \lambda^{P_{g.Z}}_{\mathbf{x}_{1:k}}(\mathbf{y}_{1:k}) = \lambda^{g.P_Z}_{\mathbf{x}_{1:k}}(\mathbf{y}_{1:k}) \quad \text{for all } \mathbf{x}_1, \dots, \mathbf{x}_k \in \mathbb{R}^n, \mathbf{y}_1, \dots, \mathbf{y}_k \in \mathbb{R}^d \tag{30}$$

$$\Leftrightarrow \frac{\lambda^P_{\mathbf{x}_{1:k},g\mathbf{x}'_{1:l}}(\mathbf{y}_{1:k}, \rho(h)\mathbf{y}'_{1:l})}{\lambda^P_{g\mathbf{x}'_{1:l}}(\rho(h)\mathbf{y}'_{1:l})} = \lambda^{P_Z}_{g^{-1}\mathbf{x}_{1:k}}(\rho(h)^{-1}\mathbf{y}_{1:k}) \quad \text{for all } \mathbf{x}_1, \dots, \mathbf{x}_k \in \mathbb{R}^n, \mathbf{y}_1, \dots, \mathbf{y}_k \in \mathbb{R}^d \tag{31}$$

$$\Leftrightarrow \frac{\lambda^P_{\mathbf{x}_{1:k},g\mathbf{x}'_{1:l}}(\mathbf{y}_{1:k}, \rho(h)\mathbf{y}'_{1:l})}{\lambda^P_{g\mathbf{x}'_{1:l}}(\rho(h)\mathbf{y}'_{1:l})} = \frac{\lambda^P_{g^{-1}\mathbf{x}_{1:k},\mathbf{x}'_{1:l}}(\rho(h)^{-1}\mathbf{y}_{1:k}, \mathbf{y}'_{1:l})}{\lambda^P_{\mathbf{x}'_{1:l}}(\mathbf{y}'_{1:l})} \quad \text{for all } \mathbf{x}_1, \dots, \mathbf{x}_k \in \mathbb{R}^n, \mathbf{y}_1, \dots, \mathbf{y}_k \in \mathbb{R}^d \tag{32}$$

where we used in row order the following facts:

1. Kolmgorov's theorem.
2. Two distributions coincide if and only if their density coincide (Lebesgue-almost everywhere).

3. Equation (24) on the left-hand side and eq. (25) on the right-hand side.
4. Equation (24) on the right-hand side.

Invariance implies equivariance. Assuming P is G-invariant, we can use eq. (27) to get

$$\lambda^P_{\mathbf{x}_{1:k}, g\mathbf{x}'_{1:l}}(\mathbf{y}_{1:k}, \rho(h)\mathbf{y}'_{1:l}) = \lambda^P_{g^{-1}\mathbf{x}_{1:k}, \mathbf{x}'_{1:l}}(\rho(h)^{-1}\mathbf{y}_{1:k}, \mathbf{y}'_{1:l}), \quad \lambda^P_{g\mathbf{x}'_{1:l}}(\rho(h)\mathbf{y}'_{1:l}) = \lambda^P_{\mathbf{x}'_{1:l}}(\mathbf{y}'_{1:l})$$

Inserting that into the left-hand side of eq. (32), we see that the equality in eq. (32) is true, i.e. $Z \mapsto P_Z$ is equivariant.

Equivariance implies invariance. By going this computation backward, we can easily show that equivariance implies invariance as well. However, there is a short-cut. Assuming that $Z \mapsto P_Z$ is equivariant, we can simply pick an empty context set $Z = \{\}$. In this case, $P_{g.Z} = P_Z = P$ and therefore equivariance implies $g.P = P$. □

B.2. Proof of theorem 1

Theorem 1. *A Gaussian process $\mathcal{GP}(\mathbf{m}, K)$ is G-invariant, equivalently the posterior G-equivariant, if and only if*

1. *$\mathbf{m}(\mathbf{x}) = \mathbf{m} \in \mathbb{R}^d$ is constant with $\mathbf{m}$ such that*

$$\rho(h)\mathbf{m} = \mathbf{m} \quad \textit{for all } h \in H \tag{8}$$

2. *K fulfils the following two conditions:*
   (a) *K is **stationary**, i.e. for all $\mathbf{x}, \mathbf{x}' \in \mathbb{R}^n$*

$$K(\mathbf{x}, \mathbf{x}') = K(\mathbf{x} - \mathbf{x}', \mathbf{0}) =: \hat{K}(\mathbf{x} - \mathbf{x}') \tag{9}$$

   (b) *K satisfies the **angular constraint**, i.e. for all $\mathbf{x}, \mathbf{x}' \in \mathbb{R}^n, h \in H$ it holds that*

$$K(h\mathbf{x}, h\mathbf{x}') = \rho(h)K(\mathbf{x}, \mathbf{x}')\rho(h)^T \tag{10}$$

   or equivalently, for all $\mathbf{x} \in \mathbb{R}^n, h \in H$

$$\hat{K}(h\mathbf{x}) = \rho(h)\hat{K}(\mathbf{x})\rho(h)^T \tag{11}$$

If this is the case, we call K ρ-equivariant.

Proof. A Gaussian process $GP(\mathbf{m}, K)$ is G-invariant if and only if

$$F \sim GP(\mathbf{m}, K) \Rightarrow g.F \sim GP(\mathbf{m}, K) \text{ for all } g \in G$$

By Kolmogorov's theorem (see Øksendal (2000)), the distribution of F and $g.F$ coincide if and only if their finite-dimensional marginals coincide. Since the marginals are normal, they are equal if and only mean and covariances are equal, i.e. if and only if

$$\mathbf{m}(\mathbf{x}) = \mathbb{E}(F(\mathbf{x})) = \mathbb{E}(g.F(\mathbf{x})) = \rho(h)\mathbf{m}(g^{-1}\mathbf{x}) = g.\mathbf{m}(\mathbf{x}) \quad \text{for all } \mathbf{x} \in \mathbb{R}^n \tag{33}$$

and for all $\mathbf{x}, \mathbf{x}' \in \mathbb{R}^n$

$$\begin{aligned} K(\mathbf{x}, \mathbf{x}') = \text{Cov}(F(\mathbf{x}), F(\mathbf{x}')) = \text{Cov}(g.F(\mathbf{x}), g.F(\mathbf{x}')) &= \text{Cov}(\rho(h)F(g^{-1}\mathbf{x}), \rho(h)F(g^{-1}\mathbf{x}')) && (34)\\ &= \rho(h)\text{Cov}(F(g^{-1}\mathbf{x}), F(g^{-1}\mathbf{x}'))\rho(h)^T && (35)\\ &= \rho(h)K(g^{-1}\mathbf{x}, g^{-1}\mathbf{x}')\rho(h)^T && (36) \end{aligned}$$

Let us assume that this equation holds. Then picking $g = t_{\mathbf{x}'}$ implies that

$$\begin{aligned} \mathbf{m}(\mathbf{x}) &= \mathbf{m}(\mathbf{x} - \mathbf{x}') \\ K(\mathbf{x}, \mathbf{x}') &= K(\mathbf{x} - \mathbf{x}', 0) \end{aligned}$$

i.e. $\mathbf{m}$ is constant and K is stationary. Similiarly, picking $g = h$ implies eq. (8) and eq. (10).
To prove the opposite direction assuming the constraints from the theorem, we can simply go these computations backwards. □

B.3. Proof of proposition 2

Proposition 2. *A conditional process model is G-equivariant if and only if the mean and covariance feature maps are G-equivariant, i.e. it holds for all $g \in G$ and context sets Z*

$$\mathbf{m}_{g.Z} = g.\mathbf{m}_Z \tag{13}$$
$$\mathbf{\Sigma}_{g.Z} = g.\mathbf{\Sigma}_Z \tag{14}$$

with $\rho_m = \rho$ and $\rho_\Sigma = \rho \otimes \rho$ the tensor product with action given by

$$\rho_\Sigma(h)A = \rho(h)A\rho(h)^T, \quad A \in \mathbb{R}^{d\times d} \tag{15}$$

Proof. Let Q_Z be the output of the model serving as the approximation of posterior distribution P_Z. It holds Q_Z is G-equivariant if and only if $Q_{g.Z} = g.Q_Z$.
If $F \sim Q_Z$, it holds by standard facts about the normal distribution

$$\begin{aligned} g.F(\mathbf{x}) =& \rho(h)F(g^{-1}\mathbf{x}) \\ \sim& \mathcal{N}(\rho(h)\mathbf{m}_Z(g^{-1}\mathbf{x}), \rho(h)\mathbf{\Sigma}_Z(g^{-1}\mathbf{x})\rho(h)^T) \\ =& \mathcal{N}(g.\mathbf{m}_Z(\mathbf{x}), g.\mathbf{\Sigma}_Z(\mathbf{x})) \end{aligned}$$

which gives the one-dimensional marginals of $g.Q_Z$. By the conditional independence assumption, $g.Q_Z = Q_{g.Z}$ if and only if their one-dimensional marginals agree, i.e. if for all $\mathbf{x}$

$$\mathcal{N}(\mathbf{m}_{g.Z}(\mathbf{x}), \mathbf{\Sigma}_{g.Z}(\mathbf{x})) = \mathcal{N}(g.\mathbf{m}_Z(\mathbf{x}), g.\mathbf{\Sigma}_Z(\mathbf{x}))$$

This is equivalent to $\mathbf{m}_{g.Z} = g.\mathbf{m}_Z$ and $\mathbf{\Sigma}_{g.Z} = g.\mathbf{\Sigma}_Z$, which finishes the proof. □

B.4. Proof of theorem 2

Theorem 2 (EquivDeepSets)**.** *Let ρ_{in}, ρ_{out} be the two fiber representations. Define the embedding representation as the direct sum $\rho_E = \rho_{triv} \oplus \rho_{in}$.*
A function $\Phi : \mathcal{Z}_{\rho_{in}} \to \mathcal{F}_{\rho_{out}}$ is G-equivariant and permutation invariant if and only if it can be expressed as

$$\Phi(Z) = \Psi(E(Z)) \tag{16}$$

for all $Z = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^m \in \mathcal{Z}_{\rho_{in}}$ with

1. $E(Z) = \sum_{i=1}^m K(\cdot, \mathbf{x}_i)\phi(\mathbf{y}_i)$
2. $\phi(\mathbf{y}) = (1, \mathbf{y})^T \in \mathbb{R}^{d+1}$.
3. $K : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^{(d+1)\times(d+1)}$ *is a ρ_E-equivariant strictly positive definite kernel (see theorem 1).*
4. $\Psi : \mathcal{F}_{\rho_E} \to \mathcal{F}_{\rho_{out}}$ *is a G-equivariant function.*

Additionally, by imposing extra constraints (see appendix B.4), we can also ensure that Φ is continuous.

Proof. This proof generalizes the proof of Gordon et al. (2020, Theorem 1).
Step 1: Injectivity of E (up to permutations).
We first want to show that under the given conditions E is injective up to permutations, i.e. $Z = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^m$ is a permutation of the elements of $Z' = \{(\mathbf{x}'_j, \mathbf{y}'_j)\}_{j=1}^{m'}$ if and only if $E(Z) = E(Z')$. By definition, $E(Z) = E(Z')$ is equivalent to

$$\sum_{i=1}^m K(\cdot, \mathbf{x}_i)\begin{pmatrix}1\\ \mathbf{y}_i\end{pmatrix} = \sum_{j=1}^{m'} K(\cdot, \mathbf{x}'_j)\begin{pmatrix}1\\ \mathbf{y}'_j\end{pmatrix} \tag{37}$$

Clearly, if Z is a permutation of Z', eq. (37) holds since one can simply change order of summands. Conversely, let us assume that eq. (37) holds. Let $f : \mathbb{R}^n \to \mathbb{R}^d$ be a function in the reproducing kernel Hilbert space (RKHS) of K (Álvarez et al., 2012). The reproducing property in the case of matrix-valued kernels says that

$$\langle f, K(\cdot, \mathbf{x})c\rangle_{\mathcal{H}} = f(\mathbf{x})^T c \quad \text{for all } c \in \mathbb{R}^d, \mathbf{x} \in \mathbb{R}^n \tag{38}$$

where $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ is the inner product on the RKHS $\mathcal{H}$. Taking the inner product with f on both sides of eq. (37), we get by the reproducing property:

$$\sum_{i=1}^{m} f(\mathbf{x}_i)^T \begin{pmatrix} 1 \\ \mathbf{y}_i \end{pmatrix} = \sum_{j=1}^{m'} f(\mathbf{x}'_j)^T \begin{pmatrix} 1 \\ \mathbf{y}'_j \end{pmatrix} \tag{39}$$

Let us choose an arbitrary $\mathbf{x}_k$ where $k = 1, \dots, m$ and let us pick $f \in \mathcal{H}$ such that $f(\mathbf{x}_k) = (1, 0, \dots, 0)^T$, $f(\mathbf{x}_i) = 0$ for all $i \neq k$ and $f(\mathbf{x}'_j) = 0$ for all $j = 1, \dots, m'$ such that $\mathbf{x}'_j \neq \mathbf{x}_k$. This is possible because K is interpolating since we assumed that K is strictly positive definite. In eq. (39), we then get

$$1 = \sum_{j=1}^{m'} 1_{\mathbf{x}'_j = \mathbf{x}_k} \tag{40}$$

Therefore, there is exactly one j such that $\mathbf{x}'_j = \mathbf{x}_k$. So every element $\mathbf{x}_k$ from Z can be found exactly once in Z'. Turning the argument around by switching Z and Z', we get that also every element $\mathbf{x}'_j$ in Z' can be found exactly once in Z. Hence, it holds that $m = m'$ and $(\mathbf{x}_1, \dots, \mathbf{x}_m)$ is a permutation of $(\mathbf{x}'_1, \dots, \mathbf{x}'_m)$. Therefore, we can now assume without loss of generality that $\mathbf{x}_i = \mathbf{x}'_i$ for all $i = 1, \dots, m$.
In eq. (39), pick now f such that $f(\mathbf{x}_i) = (0, \mathbf{y})^T$ for some $\mathbf{y} \in \mathbb{R}^d$. Then it follows that

$$\mathbf{y}^T \mathbf{y}_i = \mathbf{y}^T \mathbf{y}'_i \tag{41}$$

Since $\mathbf{y}$ was arbitrary, we can conclude that $\mathbf{y}_i = \mathbf{y}'_i$ for all $i = 1, \dots, m$. In sum, this shows that Z is a permutation of Z' and concludes the proof that E is injective up to permutations.

Step 2: Equivariance of E.
Next, we show that $Z \mapsto E(Z)$ is G-equivariant where the transformation of $E(Z)$ is defined by ρ_E as in eq. (2). Let $Z = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^m$ be a context set and $g = t_{\mathbf{x}} h \in G$. We compute

$$E(g.Z) = \sum_{i=1}^{m} K(\cdot, g\mathbf{x}_i) \begin{pmatrix} 1 \\ \rho_{\text{in}}(h)\mathbf{y}_i \end{pmatrix} = \sum_{i=1}^{m} K(\cdot, g\mathbf{x}_i)\rho_E(h) \begin{pmatrix} 1 \\ \mathbf{y}_i \end{pmatrix} \tag{42}$$

$$= \sum_{i=1}^{m} \rho_E(h) K(g^{-1}\cdot, \mathbf{x}_i)\rho_E(h)^T \rho_E(h) \begin{pmatrix} 1 \\ \mathbf{y}_i \end{pmatrix} \tag{43}$$

$$= \rho_E(h) E(Z)(g^{-1}\cdot) \tag{44}$$

$$= g.E(Z) \tag{45}$$

where the first equality follows by definition of E, the second by definition of ρ_E, the third by using ρ_E-equivariance of K, the fourth by using the assumed orthogonality of ρ_E (see section 2) and the fifth by definition.

With step 1 and 2, we can now proof the theorem.

Step 3: Universality and Equivariance of the decomposition $\Phi = \Psi \circ E$.

If $\Psi : \mathcal{F}_{\rho_E} \to \mathcal{F}_{\rho_{\text{out}}}$ is some G-equivariant function, it follows that $\Phi = \Psi \circ E$ is G-equivariant as well since it is a composition of equivariant maps Ψ and E. This shows that the composition is equivariant.

Conversely, if we assume that $\Phi : \mathcal{Z}_{\rho_{\text{in}}} \to \mathcal{F}_{\rho_{\text{out}}}$ is a G-equivariant, permutation-invariant function, we can consider it as a function defined on the family $\mathcal{Z}^{\sim}_{\rho_{\text{in}}}$ of equivalence classes of sets $Z, Z' \in \mathcal{Z}_{\rho_{\text{in}}}$ which are permutations of each other. On $\mathcal{Z}^{\sim}_{\rho_{\text{in}}}$, E is injective and we can define its inverse E^{-1} on the image of E (and set constant zero outside of the image). Clearly, it then holds $\Phi = \Psi \circ E$. Since E is equivariant, also the inverse E^{-1} is and therefore Ψ is equivariant as a composition of equivariant maps Φ and E^{-1}. This shows that this composition is universal.

This finishes the proof of the main statement of the theorem.

Additional step: Continuity of Φ. We can enforce continuity of Φ by:

1. We restrict Φ on a subset $\mathcal{Z}' \subset \mathcal{Z}_{\rho_{\text{in}}}$ which is topologically closed, closed under permutations and closed under actions of G.
2. K is continuous and $K(\mathbf{x}, \mathbf{x}') \to 0$ for $\|\mathbf{x} - \mathbf{x}'\| \to \infty$.
3. $\Psi : \mathcal{H} \to C_b(\mathbb{R}^n, \mathbb{R}^d)$ is continuous, where we denote with $C_b(\mathbb{R}^n, \mathbb{R}^d)$ the space of continuous, bounded functions $f : \mathbb{R}^n \to \mathbb{R}^d$.

The proof of this follows directly from the proof of the ConvDeepSets theorem from Gordon et al. (2020), along with the additional conditions proved above. □

C. Divergence-free and Curl-free kernels

A divergence-free kernel is a matrix-valued kernel $\Phi : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^{n\times n}$ such that its columns are divergence-free. That is $\nabla^T(\Phi(\mathbf{x}, \mathbf{x}')c) = 0 \; \forall \, c, \mathbf{x}, \mathbf{x}' \in \mathbb{R}^n$ where the derivatives are taken as a function of $\mathbf{x}$. This ensures that fields constructed by $f(\mathbf{x}) = \sum_{i=1}^N \Phi(\mathbf{x}, \mathbf{x}_i)c_i$ for some $c_i, \mathbf{x}_i \in \mathbb{R}^n$ are divergence-free. A similar definition holds for curl-free kernels.

The kernels used in this work were introduced by Macêdo & Castro (2010). In particular we use the curl- and divergence-free kernels with length scale $l > 0$ as defined for all $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^n$ by

$$K_{\text{curl}} = k_0(\mathbf{x}_1, \mathbf{x}_2)A(\mathbf{x}_1, \mathbf{x}_2), \quad K_{\text{div}}(\mathbf{x}_1, \mathbf{x}_2) = k_0(\mathbf{x}_1, \mathbf{x}_2)B(\mathbf{x}_1, \mathbf{x}_2) \tag{46}$$

where

$$k_0(\mathbf{x}_1, \mathbf{x}_2) = \frac{1}{l^2} \exp\left(-\frac{\|\mathbf{x}_1 - \mathbf{x}_2\|^2}{2l^2}\right) \tag{47}$$

$$A(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{I} - \frac{(\mathbf{x}_1 - \mathbf{x}_2)(\mathbf{x}_1 - \mathbf{x}_2)^T}{l^2} \tag{48}$$

$$B(\mathbf{x}_1, \mathbf{x}_2) = \frac{(\mathbf{x}_1 - \mathbf{x}_2)(\mathbf{x}_1 - \mathbf{x}_2)^T}{l^2} + \left(n - 1 - \frac{\|\mathbf{x}_1 - \mathbf{x}_2\|^2}{l^2}\right)\mathbf{I} \tag{49}$$

To see that K_{curl} is $E(n)$-equivariant, we compute for $g = t_{\mathbf{x}'}h \in G$

$$A(g\mathbf{x}_1, g\mathbf{x}_2) = \mathbf{I} - \frac{(h\mathbf{x}_1 + \mathbf{x}' - h\mathbf{x}_2 - \mathbf{x}')(h\mathbf{x}_1 + \mathbf{x}' - h\mathbf{x}_2 - \mathbf{x}')^T}{l^2} \tag{50}$$

$$= hh^T - \frac{h(\mathbf{x}_1 - \mathbf{x}_2)(\mathbf{x}_1 - \mathbf{x}_2)^T h^T}{l^2} \tag{51}$$

$$= hA(\mathbf{x}_1, \mathbf{x}_2)h^T \tag{52}$$

This shows that K_{curl} is $E(n)$-equivariant since k_0 is a $E(n)$-invariant scalar kernel. With a similar computation, one can see that K_{div} is $E(n)$-equivariant.

D. Experimental details

For the implementation, we used *PyTorch* (Paszke et al., 2017a). The github repository for the GP and ERA5 experiments can be found at this link and for the MNIST experiments here. The models are trained on a mix of GTX 1080, 1080Ti and 2080Ti GPUs.

To set up the SteerCNP model, we stacked equivariant convolutional layers with NormReLU activation functions in between as a decoder. The smoothing step was performed with a scalar RBF-kernel where the length scale is optimised during training. All hidden layers of the decoder use the regular representation ρ_{reg} as a fiber representation ρ of the hidden layers of the decoder if the fiber group H is C_N or D_N and the identity representation ρ_{Id} for infinite fiber groups. This choice

gave the best results and is also consistent with observations in supervised learning problems (Weiler & Cesa, 2019). For every model, we optimised the model architecture independently starting with a number of layers ranging from 3 to 9 and with a number of parameters from 20000 to 2 million. All hyperparameters were optimized by grid search for every model individually and can be found in the afore-mentioned repositories.

For the encoder E, we found that the choice of kernels K does not lead to significant differences in performance. Therefore, the results stated here used a diagonal RBF-kernel where we let the length-scale variable as a differentiable parameter. Similar to Gordon et al. (2020), we found that normalising the last d-channels with the first channel improves performance. This operation is clearly invertible and preserves equivariance.

D.1. GP experiments

For every sample we have chosen a randomly orientated grid $\mathcal{G} \subset [-10, 10]^2$ spread in a circle around the origin and sampled a Gaussian process on it with kernel K with $l = 5$. To a set of pairs $\{(\mathbf{x}, F(\mathbf{x}))\}_{\mathbf{x}\in\mathcal{G}}$, we add random noise $\epsilon \sim \mathcal{N}(0, \sigma^2)$ with $\sigma = 0.05$ on $F(\mathbf{x})$. During training, we randomly split a data set in a context set and in target set. The maximum size of a context set is set to 50. As usually done for CNPs (Garnelo et al., 2018a), the target set includes the context set during training.

D.2. ERA5 data

The ERA5 data set consists of weather parameters on a longtitude-latitude grid around the globe. We extracted the data for all points surrounding Memphis, Tennessee, with a distance of less than 520km giving us approximately 1200 grid points per weather map.

The weather variables we use are temperature, pressure and wind and we picked hourly data from the winter months December, January and February from years 1980 to 2018. Every sample corresponds to one weather map of temperature, pressure and wind in the region at one single point in time. Finally, we split the data set in a training set of 35000, a validation set of 17500 and test set of 17500 weather maps. Similarly, we proceeded for the data set from Southern China. We share the exact pre-processing scripts of the ERA5 data also in our code.

D.3. Image inpainting details

MNIST experiments

In all the experiments the context sets are drawn from $U(\frac{n_{pixels}}{100}, \frac{n_{pixels}}{2})$. We train with a batchsize of 28. The context points are drawn randomly from each batch and the rest of the pixels used as the target set. We train for 10 epochs using Adam (Kingma & Ba, 2015) with a learning rate of 3×10^{-4} for all the ConvCNP and SteerCNP models. For the CNP models we train for 30 epochs and use a learning rate of 1×10^{-3}. These values were found using early stopping and grid search respectively. Pixel intensities are normalised to lie in the range $[0, 1]$.

The dataset is additionally augmented with 10% blank images (equivalent to adding "no digit" class to the dataset in equal proportion to other classes). The rational behind this is that in the test dataset there are large regions of blank canvas. Given the model is trained on small patches, if we only trained on the MNIST digits the model would encounter these large regions of blank space, which it has never seen before. Including these blank images helped rectify this issue, and empirically led to better performance across the board. The GP lengthscale was optimised over a gird of [0.01,0.05,0.1,0.3,0.5,1.0,2.0,3.0] and the variance the same. The optimal parameters were found to be a length scale of 1.0 and a variance of 0.05.

In addition, we apply a sigmoid function to the mean prediction to ensure the predicted mean in the range $[0, 1]$. The covariance activation function is replaced with a softplus and a minimum variance of 0.01. This keeps the model equivariant as the covariance predicted is now an invariant scalar, rather than an equivariant matrix.

Table 4. Full results for the MNIST experiments. Mean log-likelihood ± 1 standard deviation over 3 random model and dataset seeds reported.

Test dataset	MNIST		rotMNIST		extrapolate MNIST		extrapolate rotMNIST	
Train dataset	MNIST	rotMNIST	MNIST	rotMNIST	MNIST	rotMNIST	MNIST	rotMNIST
Model								
GP	0.39±0.30	0.39±0.30	0.49±0.51	0.49±0.51	0.65±0.20	0.65±0.20	0.72±0.17	0.72±0.17
CNP	0.76±0.05	0.66±0.06	0.53±0.04	0.69±0.06	-1.20±0.06	-1.04±0.24	-1.11±0.06	-0.96±0.22
ConvCNP	1.01±0.01	0.95±0.01	0.93±0.05	1.00±0.05	1.09±0.03	**1.11±0.04**	1.08±0.02	1.14±0.03
SteerCNP(C_4)	1.05±0.02	**1.02±0.03**	**1.01±0.04**	**1.06±0.04**	1.12±0.02	**1.13±0.03**	1.14±0.02	**1.16±0.04**
SteerCNP(C_8)	**1.07±0.03**	**1.05±0.04**	**1.04±0.03**	**1.09±0.03**	1.13±0.01	**1.14±0.02**	**1.16±0.03**	**1.18±0.02**
SteerCNP(C_{16})	**1.08±0.03**	**1.04±0.03**	**1.04±0.08**	**1.09±0.07**	**1.14±0.04**	**1.11±0.08**	**1.17±0.05**	**1.15±0.06**
SteerCNP(D_4)	**1.08±0.03**	**1.05±0.03**	**1.04±0.01**	**1.09±0.03**	**1.12±0.05**	**1.13±0.04**	1.14±0.03	**1.17±0.06**
SteerCNP(D_8)	**1.08±0.03**	**1.04±0.04**	**1.03±0.11**	**1.10±0.06**	**1.15±0.02**	**1.12±0.02**	**1.17±0.02**	**1.17±0.02**

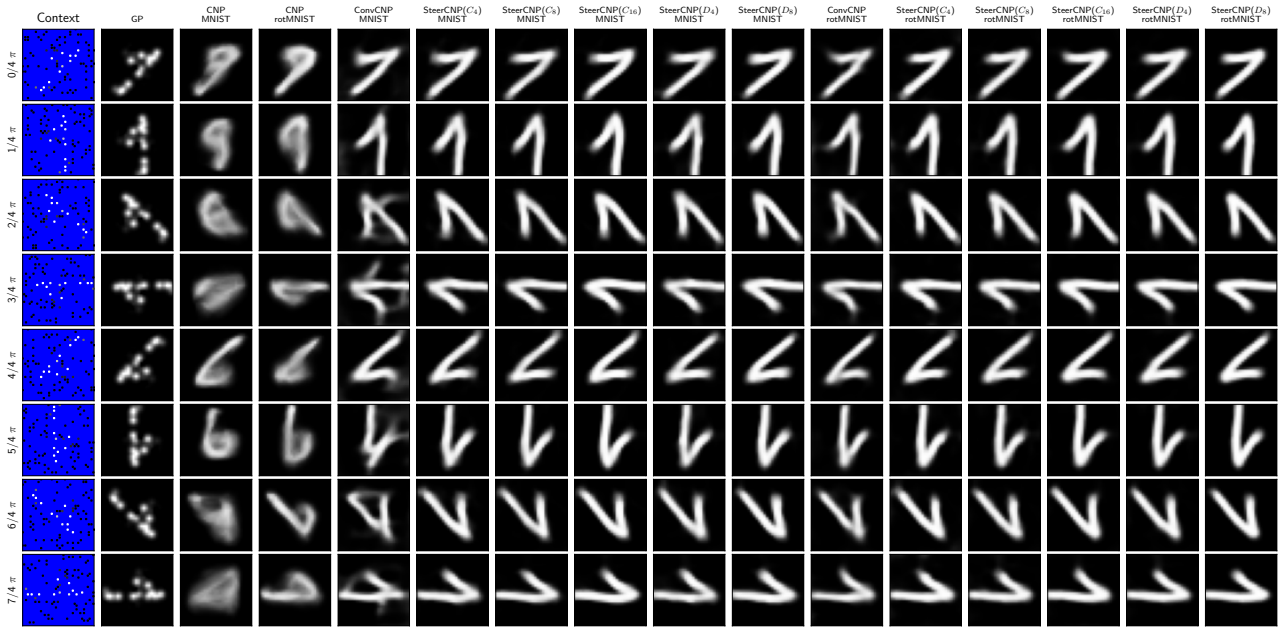

Figure 7. Qualitative examples of the behaviour of the predicted mean of different models when the context set is rotated. We observe that the ConvCNP, even when trained on rotation augmented data, has trouble predicting good shapes when the context set is rotated. By comparison the equivariant models have very consistent predictions under rotation, with the C_4 and D_4 models beng exactly equivariant to 90° rotations, and the C_{16} models being exactly equivariant to 22.5° rotations

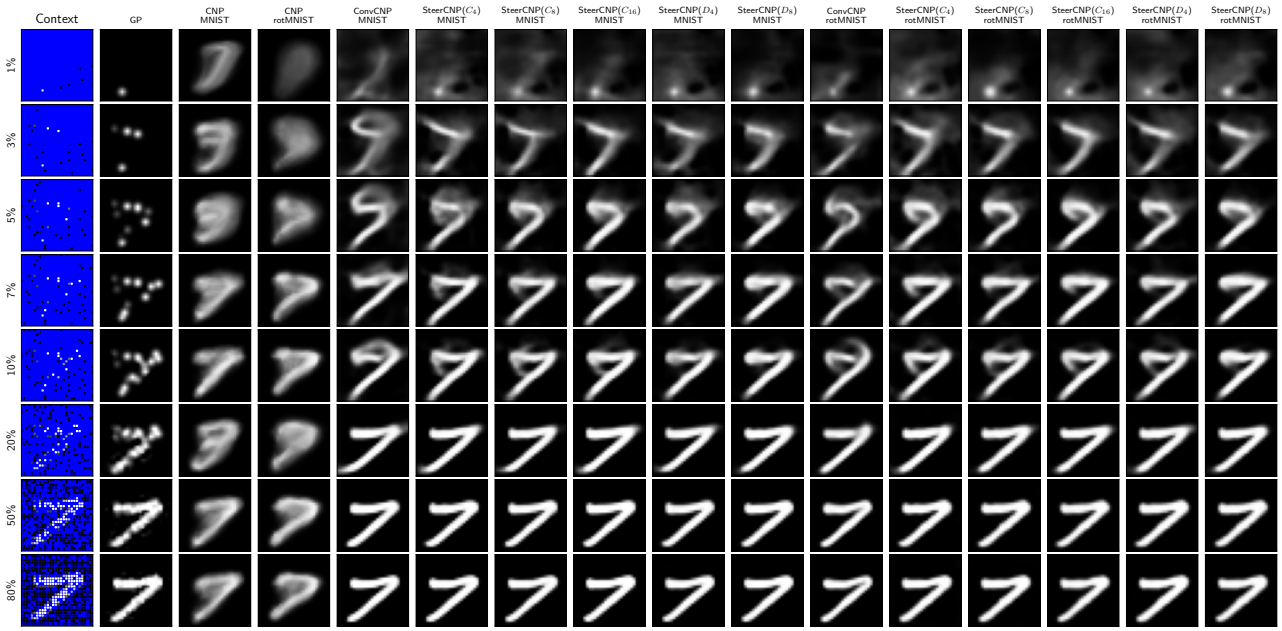

Figure 8. Qualitative examples of the behaviour of the predicted mean of different models when the context set size is changed. Digit not rotated.

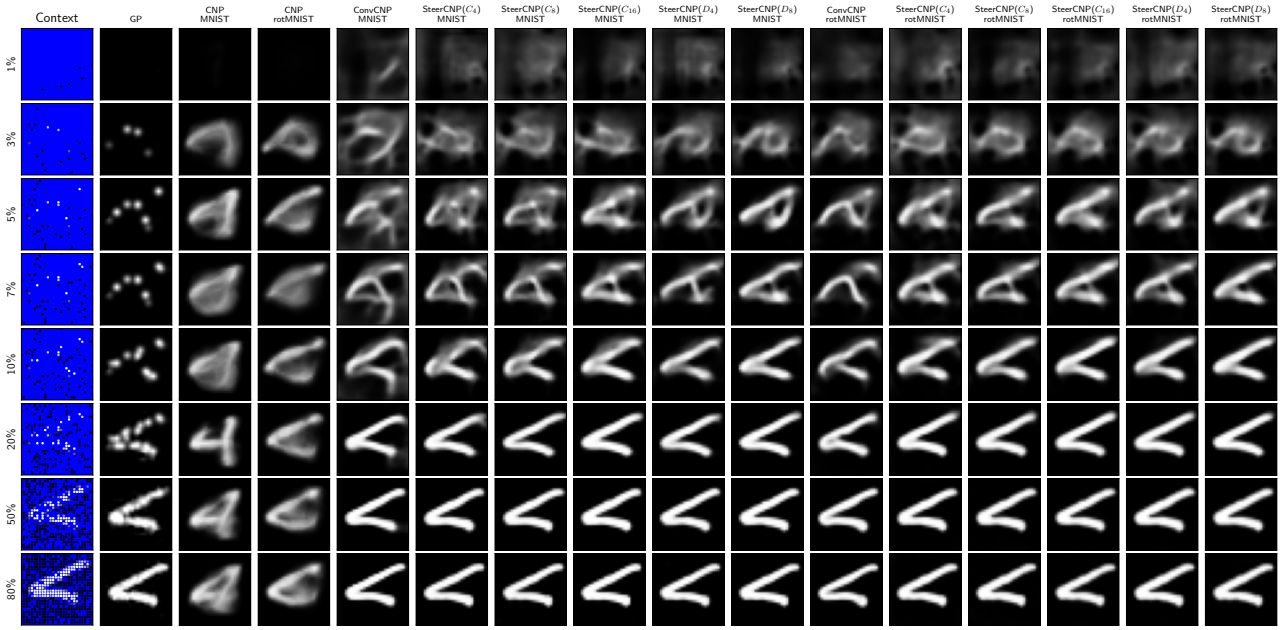

Figure 9. Qualitative examples of the behaviour of the predicted mean of different models when the context set size is changed. Digit rotated.

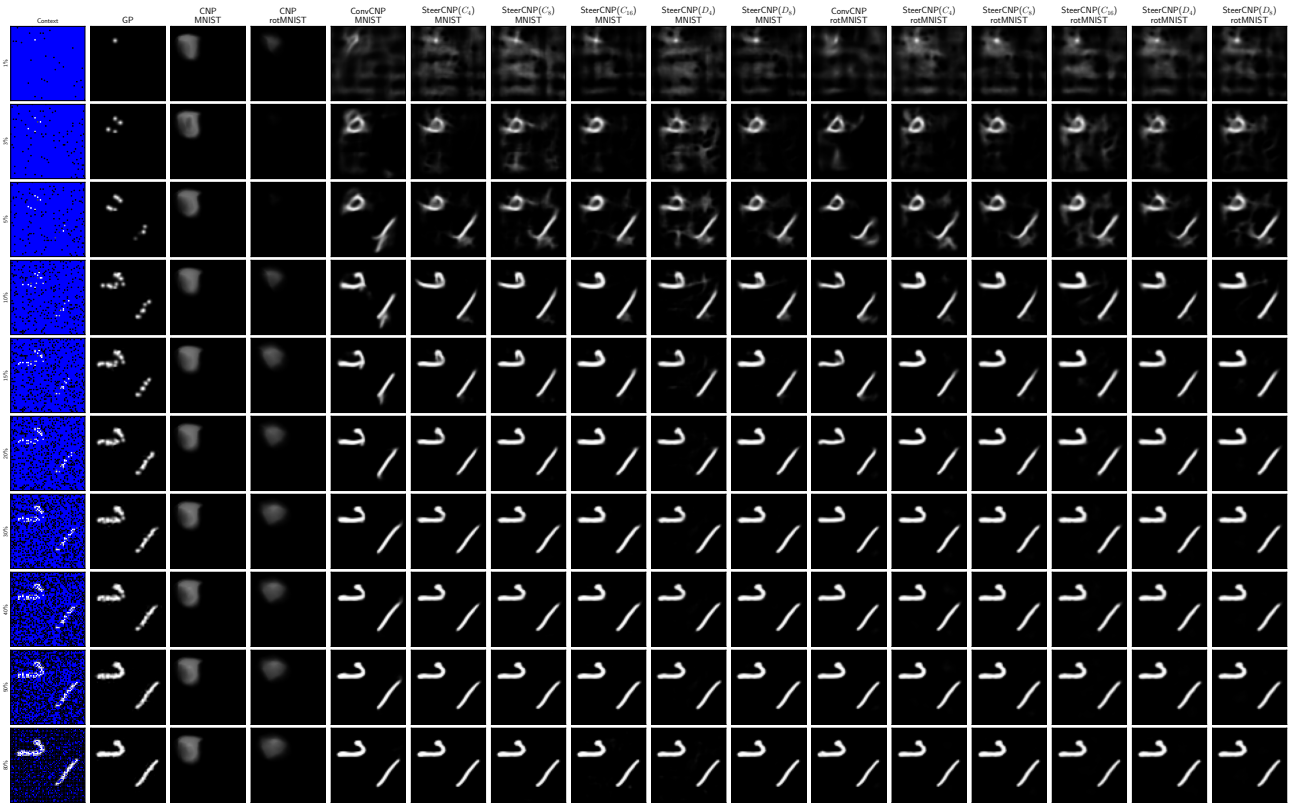

Figure 10. Qualitative examples of the behaviour of models trained on single MNIST digits, tested on multiple digits pasted into a larger canvas. Size of context set varied. We see that further away from the digits there is some noise predictions. These are likely causes by the models never having seen data as far from digits as this, leading to somewhat undefined behaviour. We see that the equivariant models exhibit considerably less of this noisy behaviour.

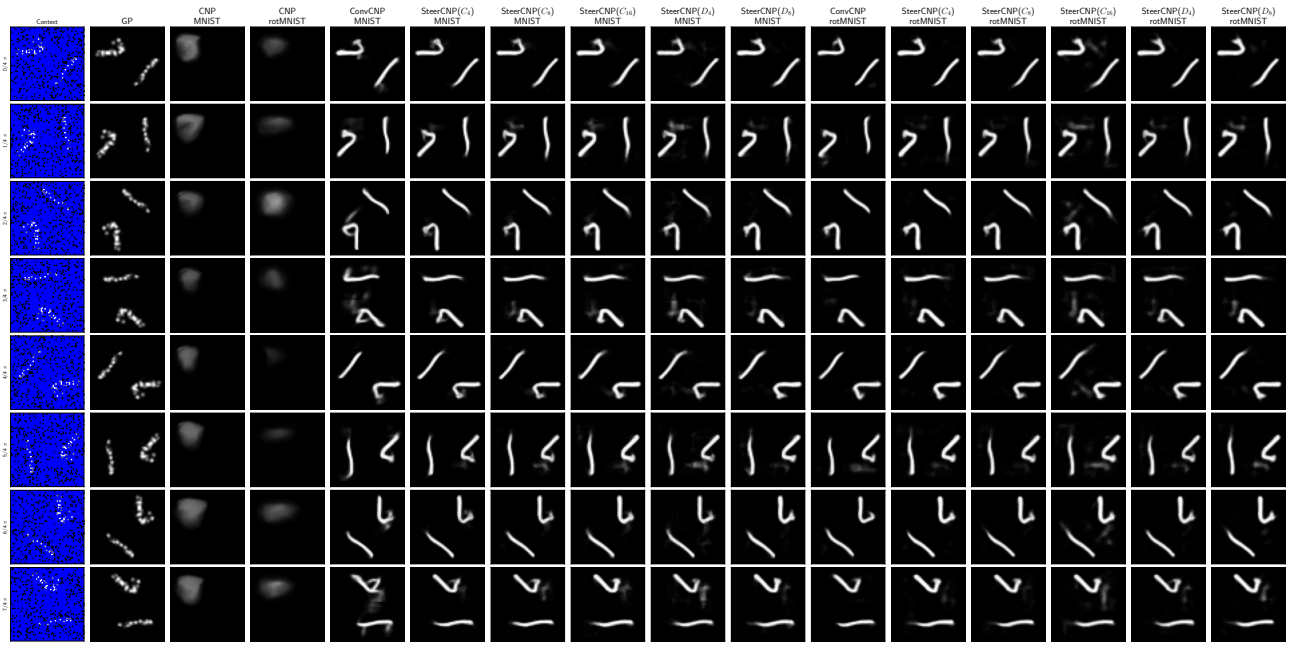

Figure 11. Qualitative examples of the behaviour of models trained on single MNIST digits, tested on multiple digits pasted into a larger canvas. Rotation of the context set is varied. We see that the equivariant models show very little change in behaviour under rotation, whereas the ConvCNP gives reasonably wild predictions under rotation.